

PROGRAM MANAGER

Journal of the Defense Systems Management College

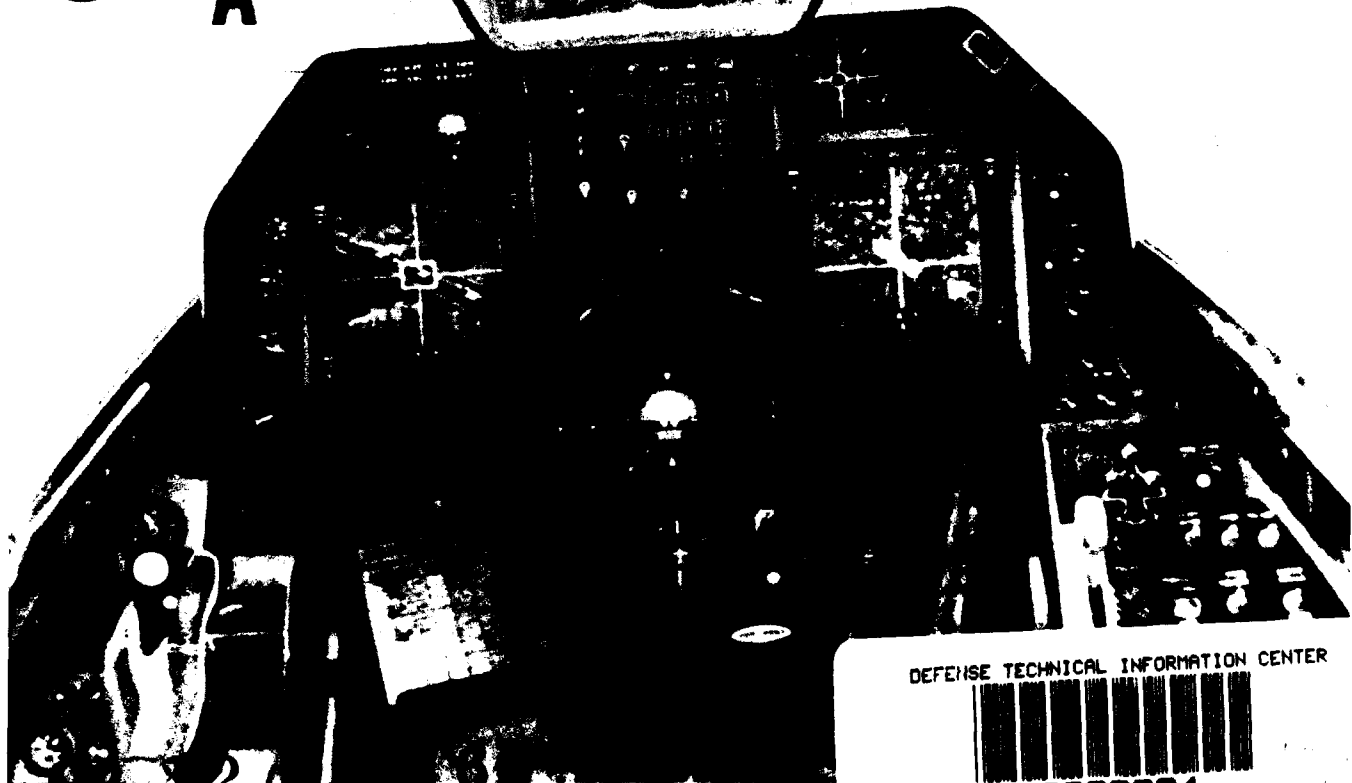
Path to Quality

Cost Avoidance

Lessons Learned

Real Time Software Manipulation for Weapons Systems

S DTIC
ELECTE
DEC 28 1992
A **D**



DEFENSE TECHNICAL INFORMATION CENTER



9232804

This document has been approved
for public release and sale; its
distribution is unlimited.

PROGRAM MANAGER

Journal of the Defense Systems Management College

VOL. XXI, No. 6, DSMC 111



DEFENSE SYSTEMS MANAGEMENT COLLEGE

Commandant
Rear Admiral William L. Vincent, USN

Provost
Gregory T. Wierzbicki

Executive Director
Research and Information Department
Lt Col Carl Bryant, USAF

Director of Publications
Robert W. Ball

PROGRAM MANAGER

Managing Editor
Catherine M. Clark

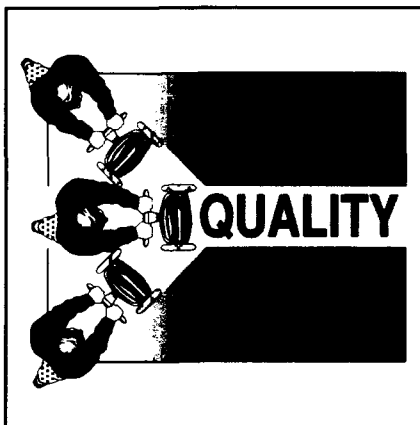
Associate Editor
Esther M. Farria

Art Director
Greg Caruth

Typography and Design
Paula Croisetiere

Program Manager (ISSN 0199-7114) is published bimonthly by the Defense Systems Management College, Fort Belvoir, VA 22060-5426. Nongovernment employees and organizations may subscribe at \$9.50 annually through the Superintendent of Documents, U.S. Government Printing Office, Washington, DC 20402. Second class postage paid at Fort Belvoir, VA, and at additional entry offices.

POSTMASTER: Send address changes to *Program Manager*, Defense Systems Management College, Fort Belvoir, VA 22060-5426.

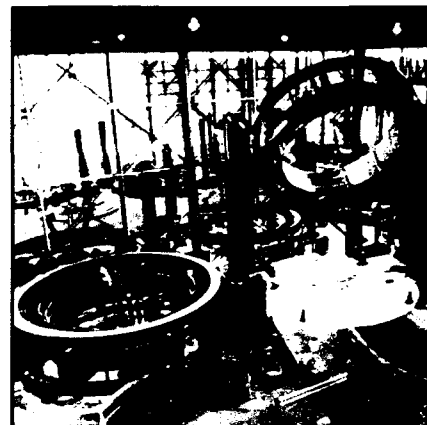


2

Cutting a Path to Quality through Concurrent Teamwork

William N. Maher

Remaining alive and competitive.

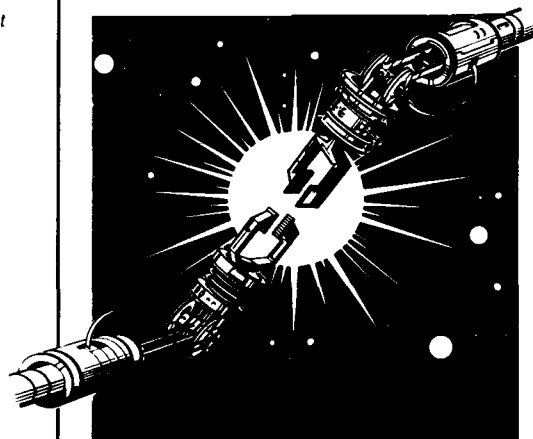


8

Manufacturing Notes

William T. Motley

Tooling impacts on cost, quality, production rate and product uniformity.



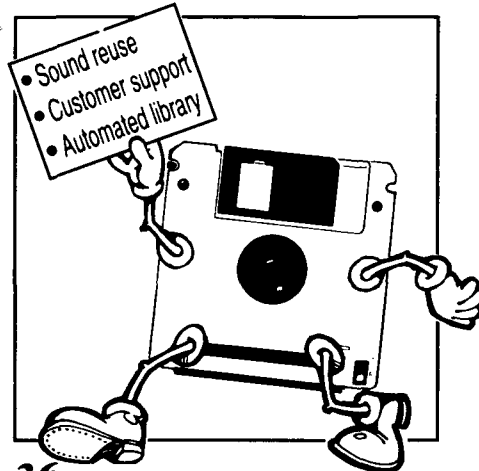
28

Lessons Learned in Manufacturing Modernization

Dr. Robert E. Schafrik

Dr. Thomas D. Fiorino

Best practices and procedures.



36

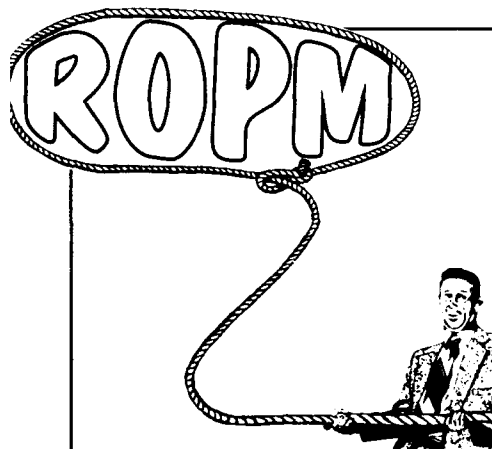
The Army Reuse Center—Cost Avoidance through Software Reuse

Marrea Riggs

Software development support for program managers.

*For sale by the Superintendent of Documents, U.S. Government Printing Office
Washington, DC 20402*

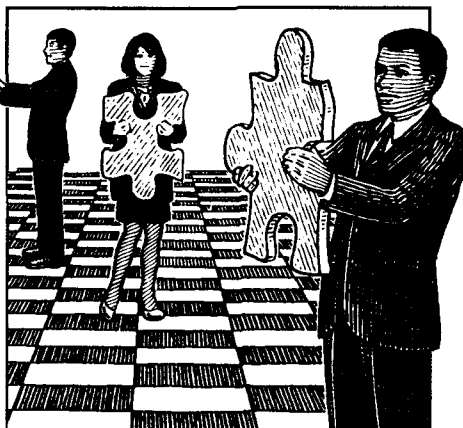
Whenever in this publication "man," "men," or their related pronouns appear, either as words or parts of words (other than with obvious reference to named male individuals), they have been used for literary purposes and are meant in their generic sense.



12 Results-Oriented Program Management as a Leadership/Management Model

James E. Price
Mary-Blair Valentine

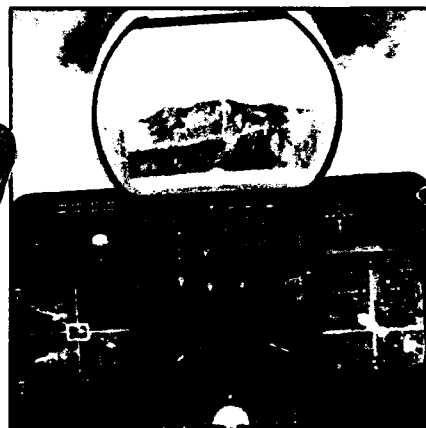
Mandating careful stewardship of resources and delivering a product satisfying to the customer.



18 Organization Development

Major Jim Wilson, USA

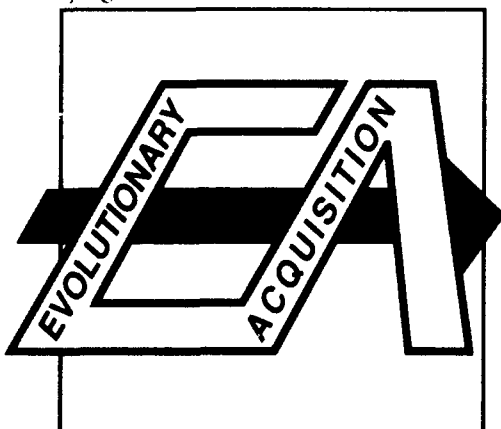
What Is It? Can You Use It?



22 The Software Development Process

Captain Jeanne L. Murtagh, USAF

An integral part of weapon systems.



38 Evolutionary Acquisition and DOD's Revised Acquisition Strategy

Edward Hirsch Brigadier General, USA
(ret.)

The Defense Systems Management
College—DOD's agent for change.

ALSO:

Annual Reliability and
Maintainability
Symposium

10

Foreign Language inside back
cover

| | |
|--------------------|-------------------------------------|
| Accession For | |
| NTIS CRA&I | <input checked="" type="checkbox"/> |
| DTIC TAB | <input type="checkbox"/> |
| Unannounced | <input type="checkbox"/> |
| Justification | |
| By 2.30 per isme | |
| Distribution/ GPO | |
| Availability Codes | |
| Dist | Avail and/or Special |
| A-1 | 24 |

DTIC QUALITY INSPECTED 5

Program Manager is intended to be a vehicle for the transmission of information on policies, trends, events, and current thinking affecting program management and defense systems acquisition.

Statements of fact or opinion appearing in *Program Manager* are solely those of the authors and are not necessarily endorsed by the Department of Defense or the Defense Systems Management College. Unless copyrighted, articles may be reprinted. When reprinting, please credit the author and *Program Manager*, and forward two copies of the reprinted material to the Director of Publications.

To subscribe, government personnel should submit written requests (using their business addresses) to the Director of Publications.

Manuscripts and other correspondence are welcome and should be addressed to the Director of Publications. Inquiries concerning proposed articles may be made by phone at (703) 805-2892/3056 or DSN 655-2892/3056.

CUTTING A PATH TO QUALITY THROUGH CONCURRENT TEAMWORK

Remaining Alive and Competitive

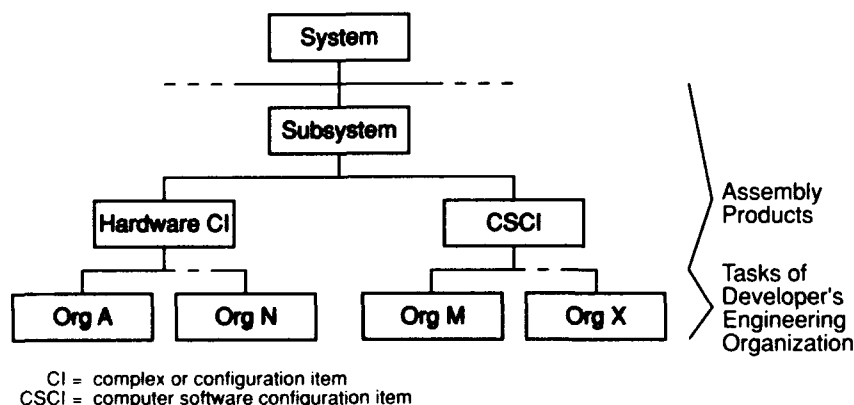
William N. Maher

Now that the robust defense market has been permanently weakened by seismic geopolitical shifts, system developers wanting to remain alive and competitive must drastically improve the quality of their products, and find ways to design and build them faster and more efficiently. Engineering in particular must improve its competitiveness by promoting teamwork, concurrent processes and total quality management. In this paper, I will discuss how focusing on teamwork, concurrent processes, and quality can be facilitated by structuring all work according to the product concept of MIL-STD-881B, and explicitly incorporating in the work breakdown

structures (WBS) a DoD-STD-2167A-derived process. The key is to structure the work of engineering to promote quality in the engineering process and by extension into the system being developed.

Integrating work organization based on MIL-STD-881B with the workflow decreed by DoD-STD-2167A can help bring about desired changes. Many defense contractors have tended to misconstrue 881B by applying its work organizing techniques to the performing department rather than to the system being developed. The resulting departmental work breakdown structures have assumed rather than ex-

FIGURE 1. Organizational Orientation or Incomplete Product Decomposition



Mr. Maher is Equipment Technology Manager, C3 Systems, GTE Government Systems Corporation, Needham Heights, Mass.

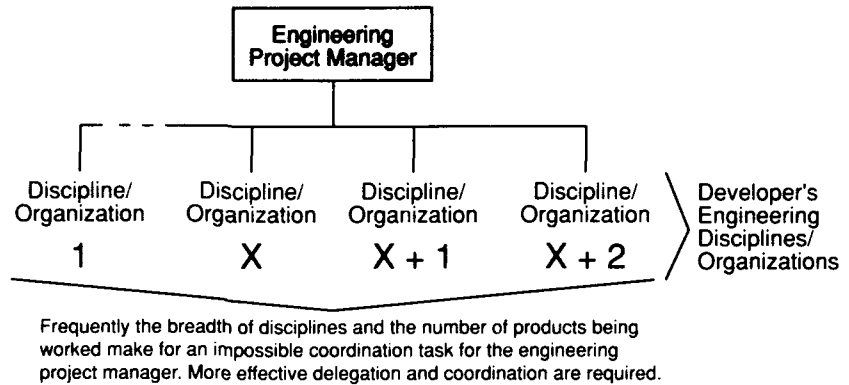


explicitly specified 2167A's workflow for lower-level products like printed wiring assemblies (PWA) and software components.

Emphasizing the Organization Doesn't Work

Typically, engineering decomposes a system into subsystems which are, in turn, decomposed into hardware configuration items (Hardware CIs) and computer software configuration items (CSCIs). Each subsystem and CI contains lower-level products that go through a development process consisting of requirements analysis, design, build and test. Historically, at this point system decomposition and

FIGURE 2. Organization/Discipline Coordination or Inappropriate Delegations



work structuring (to say nothing of cost accounting) shifts from the architecture, or system products, to the tasks performed by various engineer-

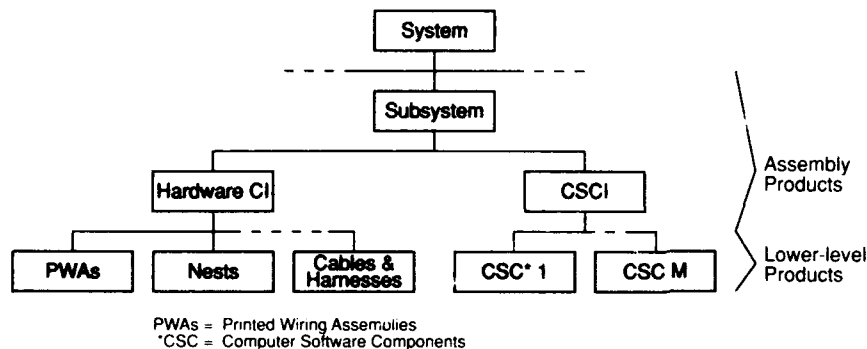
ing disciplines developing the products. This shift, depicted in the lower half of Figure 1, has at least three adverse consequences:

—It fails to define explicitly the lower-level hardware and software products needing requirements analysis, design, build and test. Too often, the engineering project manager is forced to make this process happen.

—It creates a "stovepiped" rather than a team approach, resulting in inefficiency and ineffectiveness in the engineering process and, ultimately, degrades the quality of product. Figure 2 shows the work defined in terms of stovepiped disciplines or departments. The engineering project manager is directed to achieve a team effort by coordinating these disciplines or, worse, it is assumed the disciplines will communicate and coordinate themselves. When the scope of coordination is large in terms of disciplines and multidisciplinary tasks, delegating responsibility to one person is inappropriate and risky. Furthermore, expecting individual disciplines to coordinate themselves is unrealistic because their focus is too narrow; each concentrates on a single requirement rather than on the complete set of requirements. In effect, inappropriate delegations fail to exploit the real expertise in such major engineering functions as systems for systems products, software for software products, and equipment for equipment products.



FIGURE 3. Product Orientation or Complete Product Decomposition



—It results in poor processes—stovepiped functions, serial activities with “over-the-wall” handoffs, and confusion of roles and responsibilities. Further, it obstructs helpful work and process styles like multifunction, concurrent, and project-focused processes, teamwork, and clear delegations of responsibility, all of which must be encouraged if performance objectives are to be met expeditiously and efficiently.

Shifting Emphasis Product

How do you shift work orientation to lower-level products and what, if anything, does it change? Figure 3 shows decomposition of a system into hardware and software CIs and then into the lower-level hardware and software products of those CIs. Each product needs a requirements analysis, design, build and test process multifunction and multidepartment team processes. When engineering

decomposes a system to structure its work in accordance with 881B, the work breakdown structure consists entirely of products instead of organizations, which is truer to the letter and spirit of 881B. By taking this approach, engineering becomes product-oriented rather than department-oriented. This has three major advantages.

—It explicitly acknowledges the complexity of lower-level products and implicitly acknowledges the need for requirements analysis, design, build and test for each one.

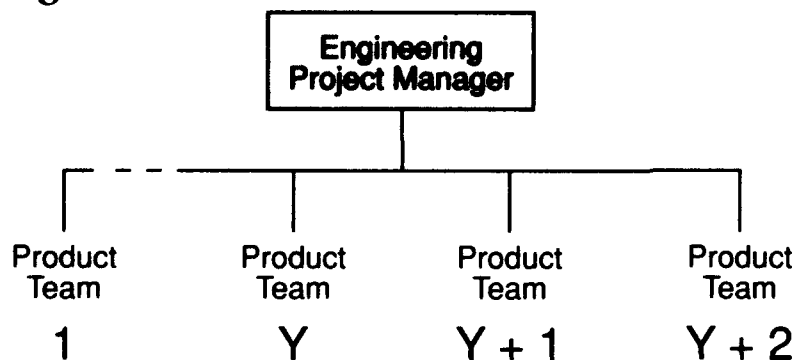
—It promotes concurrent engineering by focusing attention where it ought to be focused—on designing the product rather than on individual tasks performed by the department.

—It favors a team approach because it defines cost accounts by product rather than organization. This

makes team leaders responsible for the quality of their respective products, as opposed to the engineering project manager, who then delegates responsibilities to team leaders. They, in turn, coordinate disciplines, as shown in Figure 4. To put it another way, the engineering project manager coordinates teams rather than disciplines, and management interfaces are between the engineering project manager and leaders of multidiscipline teams. The flowdown of responsibility follows the hierarchy of the developed system, as shown in Figure 5, instead of the organization chart. In this example, responsibility for software and equipment engineering for software and equipment correctly flows through systems engineering. To demonstrate advantages of product orientation, consider the systems engineering function. Difficulties often are attributed to emergence of individual engineering specialties and obstacles they pose to coordination, integration and teamwork throughout the development process. Product orientation helps systems engineering team (or product) leaders focus on integrating all the specialties; system product leaders have clear responsibility to coordinate and delegate the work to various specialties involved in creating the system products.

—It clarifies goals and roles, substitutes product goals for departmental or discipline goals, and replaces failure-prone with success-oriented styles of working.

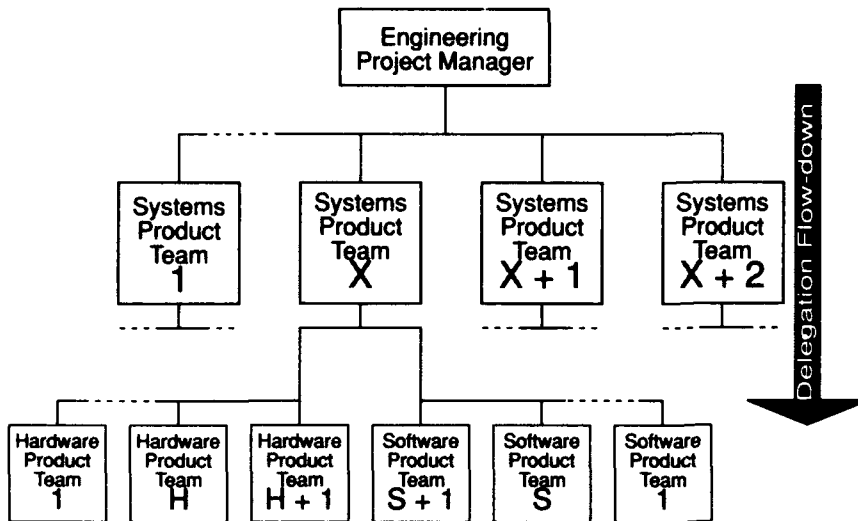
FIGURE 4. Team/Product Coordination of Effective Delegation



Phased Product Orientation

To make explicit the requirements analysis, design, build and test processes for lower-level system products, engineering must take product orientation a step further. It must divide systems, hardware, and software products along the lines of the subphases of a 2167A-derived process for engineering and manufacturing development (EMD), which is the same as full-scale development. This is another way of saying engineering

FIGURE 5. Hierarchal Team Coordination



must embed activities of the 2167A process into the work breakdown structure.

The products of the 881B WBS go through a multiphase sequence of activities in accordance with 2167A, during which they evolve from one appearance to another; each phase's product feeds the next phase as an input and, during each phase, specific activities cause the transition from inputs to outputs. Figure 6 illustrates that emphasizing product instead of procedure in any phase is equivalent to emphasizing outputs rather than means of getting those outputs. In other words, product orientation emphasizes results rather than the means for getting those results. Outputs and inputs outside the box in Figure 6 are emphasized rather than the procedures inside the box. While input/output values provide stable planning parameters, the means to achieve them may be changed and improved.

Figure 7 shows the phases of a sequential or waterfall DOD-STD-2167A-derived development process for systems, hardware and software. Figure 8 shows that combining 2167A with 881B results in multiple phases for each product, creating a dynamic WBS. In this dynamic WBS the front-end system design, system design implementation, and system integra-

tion and test process for subsystems are now explicit rather than buried within a static WBS. In addition, the requirements, design, build, and test process for hardware and software CIs and their lower-level products are also explicit. This dynamic WBS establishes:

- A project work environment for applying internal vendor/customer concepts
- Continuously improvable yet "constant" processes
- Statistical process control of engineering design processes
- Project cost estimates based on products being developed.

Internal Vendor Must Satisfy Internal Customers First. Products often

become the responsibility of another internal function for the activities of the next phase of the project process. For example, the build-to-design Printed Wiring Assembly (PWA) product goes from equipment engineering to manufacturing. Consequently, the phased product orientation allows internal vendors to satisfy needs of internal customers. The internal vendor, in this case equipment engineering, must include on the team its internal customer, manufacturing. The quality implications are significant. Satisfying needs of internal customers goes a long way toward achieving world-class quality products for external customers.

Catch 22. How to continually improve engineering processes, yet retain a basic stability, is the "Catch 22" we confront today. Although components of a product in each of its phases are the same, the procedures by which they are assembled may be continuously improved. For example, the build-to-design phase of a PWA contains a stable set of components: printed wire layouts, drill tapes, auto insertion tapes, assembly design, etc. The procedures for generating these components can be improved by introducing computer-aided techniques, resulting in stability of the workflow along with continuous improvement of procedures.

Applying Statistical Process Control to Engineering Design. Given the stability of product inputs and outputs in each phase, it is possible to schedule a sequence of inputs and outputs

FIGURE 6. Phased Products and Input/Output Emphasis

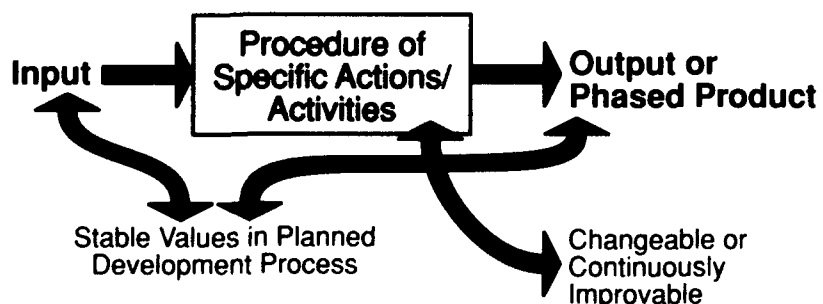
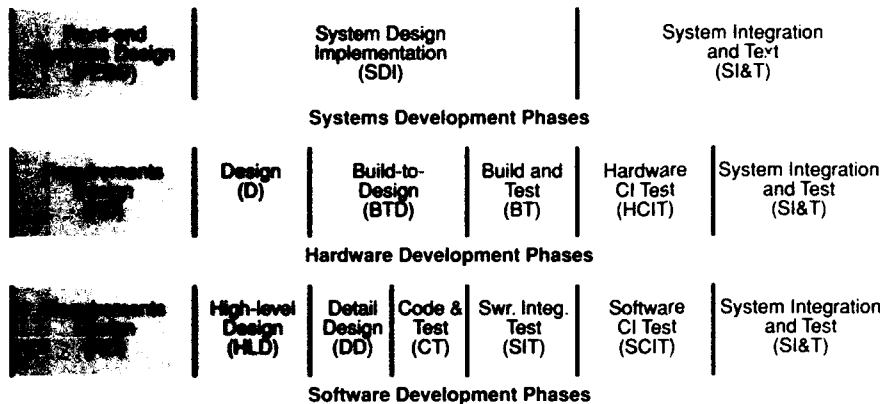


FIGURE 7. Subphases of Engineering and Manufacturing Development



for the engineering design process. Variability in the value of these components, such as an incomplete drill-tape for a PWA, is a deviation in the process which degrades quality. Engineering, like manufacturing, can strive to control variations in its design and development process and thereby achieve greater quality.

Estimating Project Cost by Product Rather than Engineering Discipline. Phased-product orientation opens a

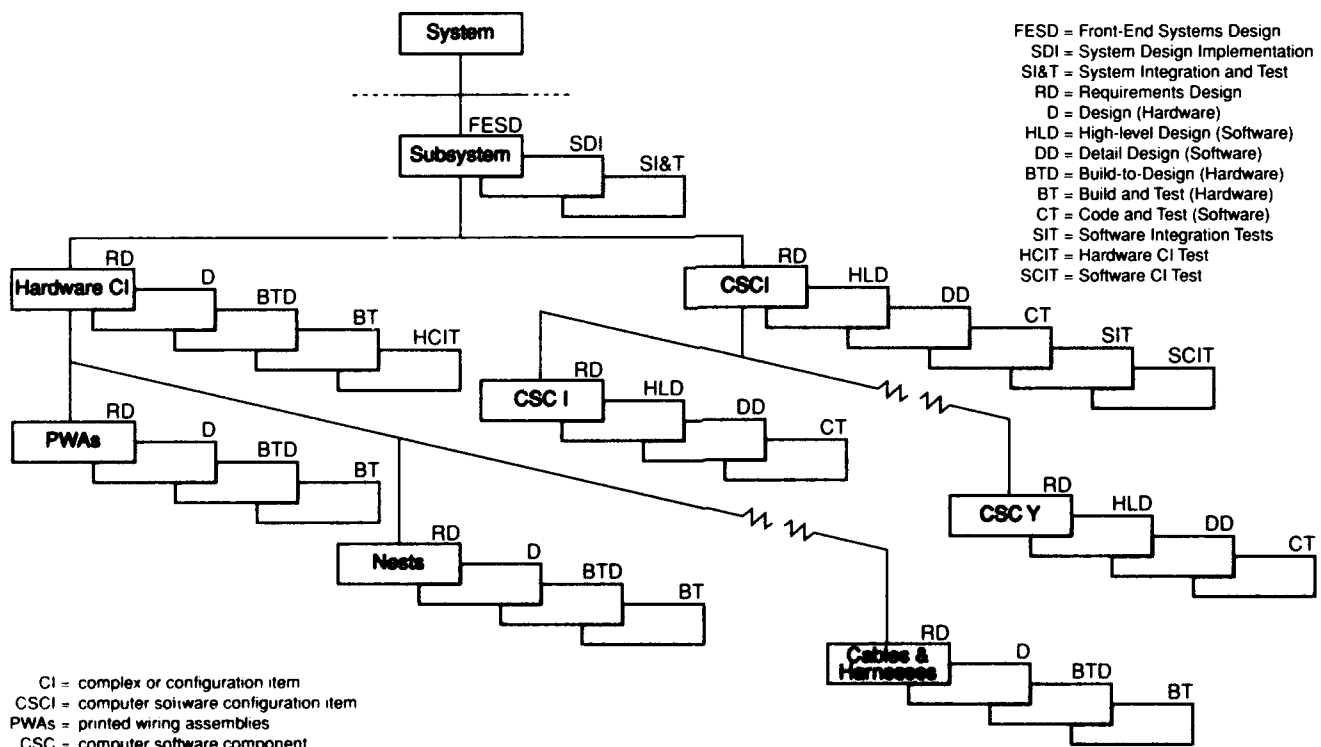
way to control variability in the estimating process and sets the stage for estimating engineering costs according to the products required to build the system and the phases of the process, rather than according to engineering disciplines or departments that will do the work. On ensuing projects, phased-product cost accounts permit collecting historical cost data for phased-product tasks. These may be used to estimate new projects cost and set continuous improvement goals.

A significant step affecting engineering performance is defining products that will make up the system. A good many engineering shortfalls occur when the number of products increases beyond what was estimated during the proposal stage. Phased-product orientation allows these variations to be tracked and controlled.

Conclusion

In conclusion, we see that integrating 881B and 2167A is the route to better teamwork, concurrent rather than serial engineering workflows, and improved product quality for the lower-level products and, by extension, the entire system being developed. The best way to implement this integration is by shifting the emphasis from the departments performing the work to the system and its products.

FIGURE 8. Phased Product WBS



CALL NOW!

**THE DEFENSE SYSTEMS
MANAGEMENT
COLLEGE**

HAS

**INDUSTRY
STUDENT
VACANCIES**

***TUITION
IS WAIVED
FOR
ELIGIBLE
STUDENTS.***

***LEARN THE
ACQUISITION
PROCESS
FROM THE
BUYER'S
SIDE.***

**FOR
PROGRAM
MANAGEMENT
COURSE 93-1**

25 JANUARY - 11 JUNE 1993



LINDA STILTNER

MANUFACTURING NOTES

Tooling

William T. Motley

Tooling includes devices like cutters, jigs, fixtures, templates and worker aids that are required to form, shape, fabricate, assemble, hold, move or handle prime equipment or any part of it. Tooling is used to repeatedly reproduce components under production conditions. Tooling has impacts on cost, quality, production rate and product uniformity. Tooling plays a major role in producing products made out of metal or composites.

Accurate tooling results in the interchangeability of parts. Attaining uniform product (minimum variation from a dimensional specification) is one of the primary purposes of tooling. Dimensional accuracy and rigidity of clamping are the prime requirements of good tooling. In addition to interchangeability, tooling also is used by workers to make their work easier and faster with less chance of error.

Tooling can be broken down into four broad areas:

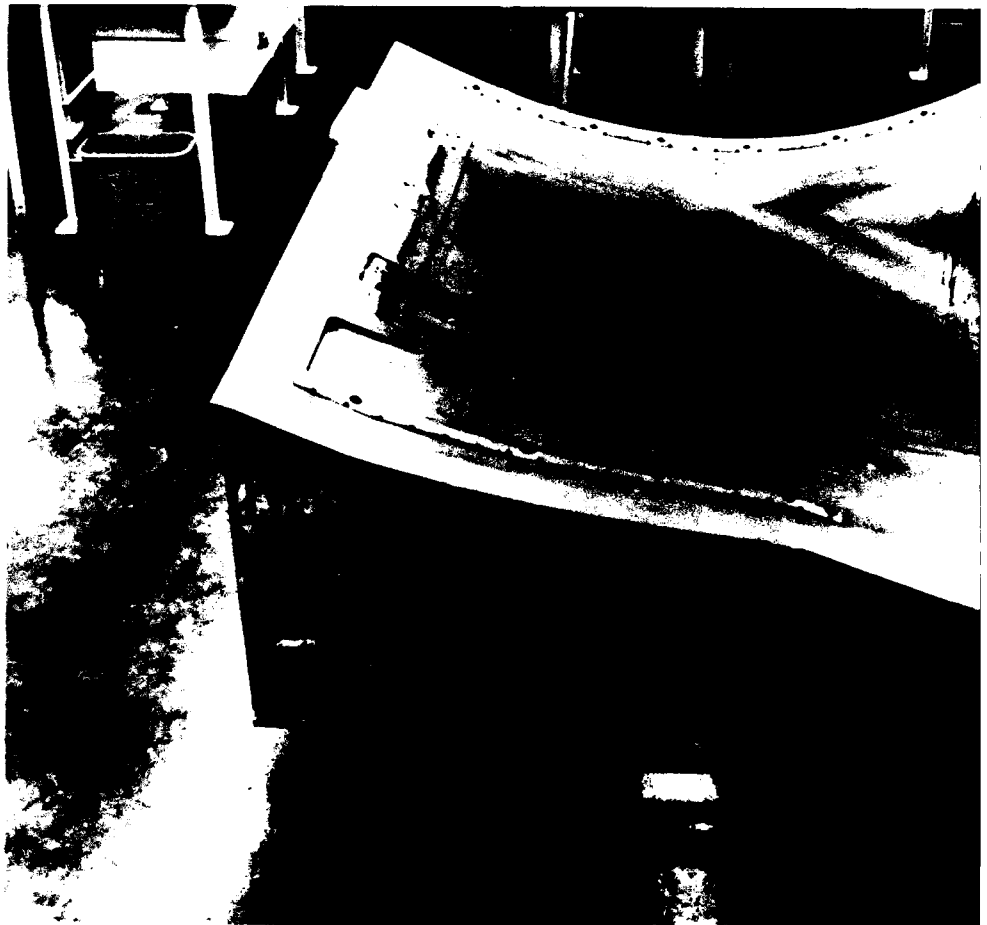
—Cutting Tools (perishable); used to machine detail parts to shape and to generate holes. Mills, drill bits and reamers are examples.

—Machine Fixtures; hold detail parts in position while processing operations take place. Drill fixtures and welding fixtures are examples.

—Detail Part Tools; fabricate parts from raw stock or continue the processing of parts that were processed on other detail part tools. Lay-up tools for composite structures are an

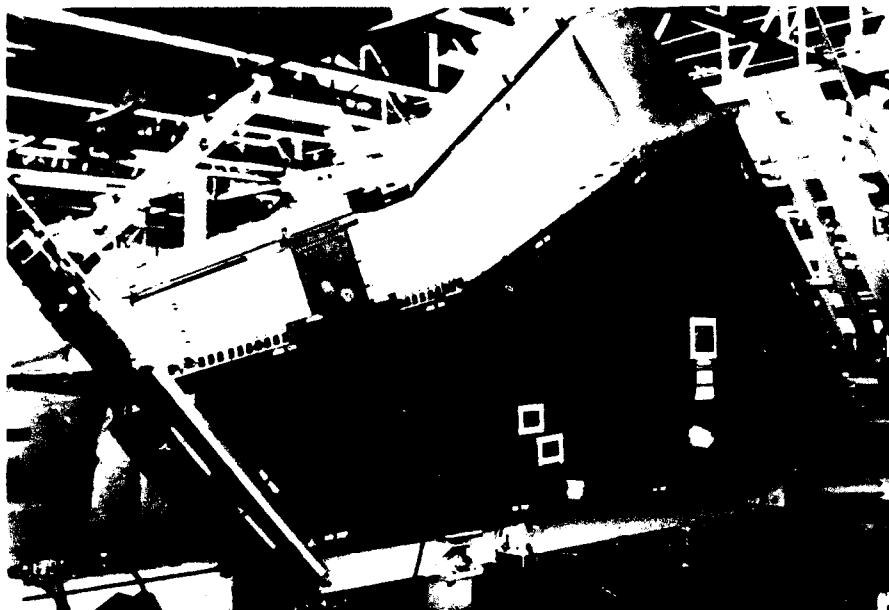
example of this type of tooling. Templates and casting molds also fall into this category.

—Assembly Tools; assembly tools are usually the largest of all tools and frequently the most complex. When two or more parts are to be accurately located and attached to each other an assembly tool usually is needed.



Bond jig used in the construction of composite aircraft parts.

Mr. Motley is the chair of the Manufacturing Management Department, Defense Systems Management College.



Aircraft canopy assembly fixture.

Assembly tooling can be hundreds of feet long and many stories high, depending on how many detail parts are to be assembled together and the size of the finished product. A wing assembly jig holds spars, ribs, and

wing skins relative to each other so they may be riveted together after drilling.

Good Tooling Practices

—Minimize set-ups and breakdowns of any manufacturing process. Strive for single fixture operations—changing set-up and fixtures introduces variability into the manufacturing process.

—Practice functional gaging and tooling; i.e., gages

and tools should create the same physical interfaces and distortions on critical surfaces as do the actual mating components.

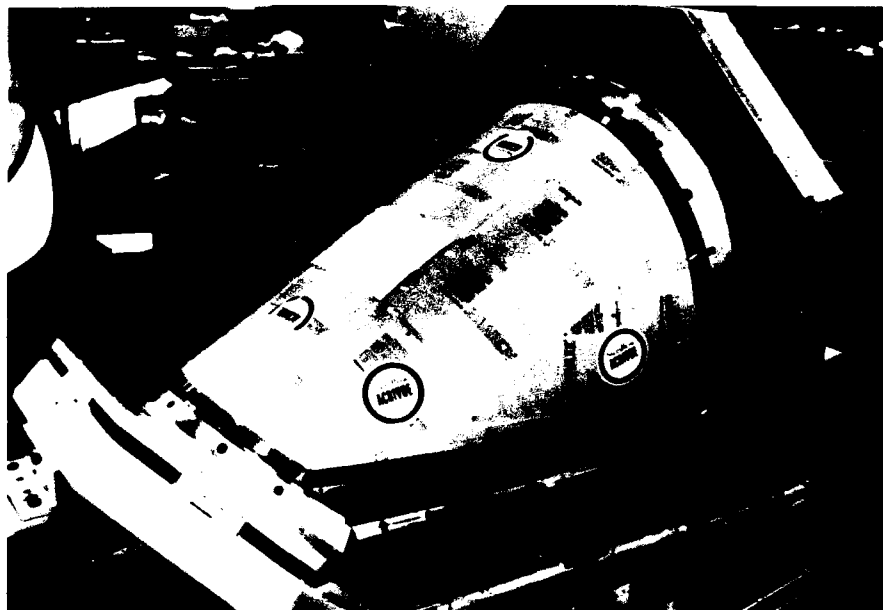
—To minimize variability, gage and inspect parts while in the fixture.

—Prime contractors should provide their subcontractors and suppliers with layout drawings, detail drawings and photographs of all critical manufacturing and acceptance tooling. To ensure uniform tooling philosophy and practices, a firm requirement for plans, reviews, and demonstrations must be

in each subcontractor's statement of work, and timely reviews of the subcontractor's tooling efforts must be made by the prime contractor.

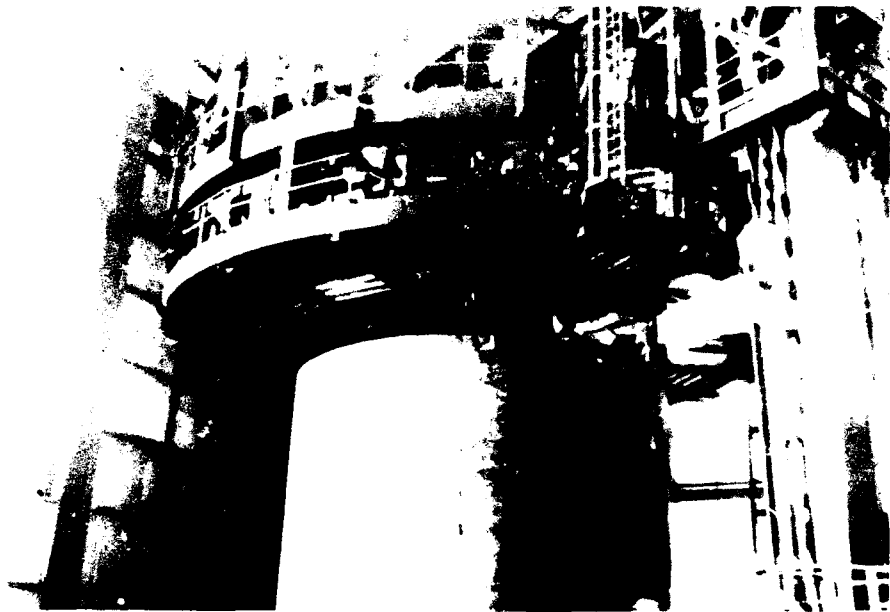
—For minimum variability, all design, manufacturing and tooling activities should be based upon a Geometric Dimensioning and Tolerancing drawing system. Such a drawing system bases its geometric datums upon actual physical attributes of a com-

For minimum variability, all design, manufacturing and tooling activities should be based upon a Geometric Dimensioning and Tolerancing drawing system.



Canopy Assembly fixture.

**On a
typical
aircraft contract,
there are as many
or more tools
as there are
parts.**



Fixtures and jiggig used in the construction of submarine hulls.

ponent rather than imaginary datums such as centerlines.

—To minimize variability and maintain configuration control, all master/gage tooling should be produced by the same source.

Program Office Concerns

—Tooling design and fabrication is a complex, specialized field. The difficulty, cost and time requirements

of a major tooling effort should not be underestimated.

—Tooling planning must be initiated simultaneously with the manufacturing plan.

—Tooling must be periodically calibrated; it is subject to wear and misalignment.

—Tooling should be discussed at technical design reviews.

—Tooling wears out; provisions must be made to finance the replace-

ment of tooling during the life of a program.

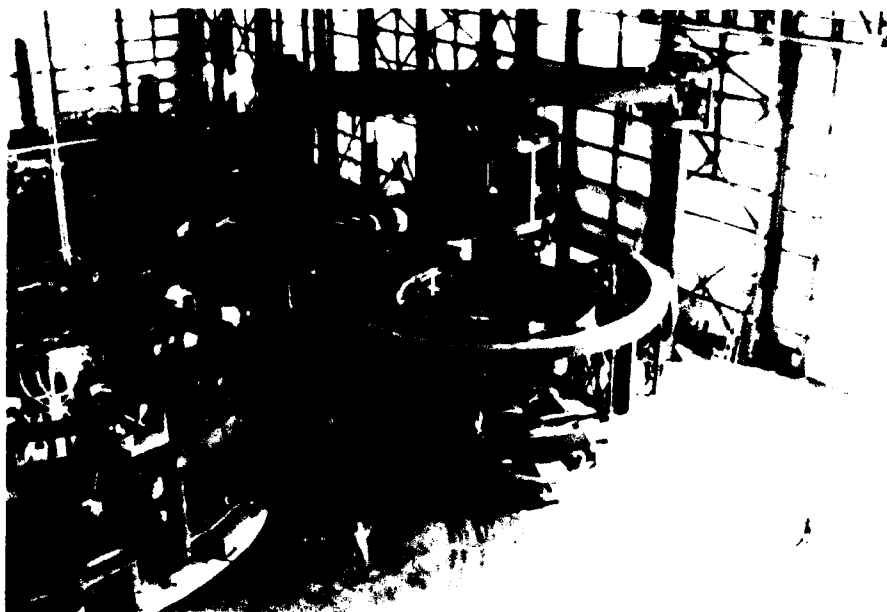
—General tools planning guidelines need to be established for contractor investment, the level of rate tooling to be utilized, and the transition from limited life to rate tooling.

—Prime contractors and subcontractors must have and execute a tooling control plan that addresses tooling inventory control, tooling configuration control and tooling maintenance and calibration. On a typical aircraft contract, there are as many or more tools as there are parts. All of these tools must be cataloged, numbered, and stored and kept up-to-date as design and manufacturing changes occur. It is apparent that money can be saved by eliminating unneeded tools.

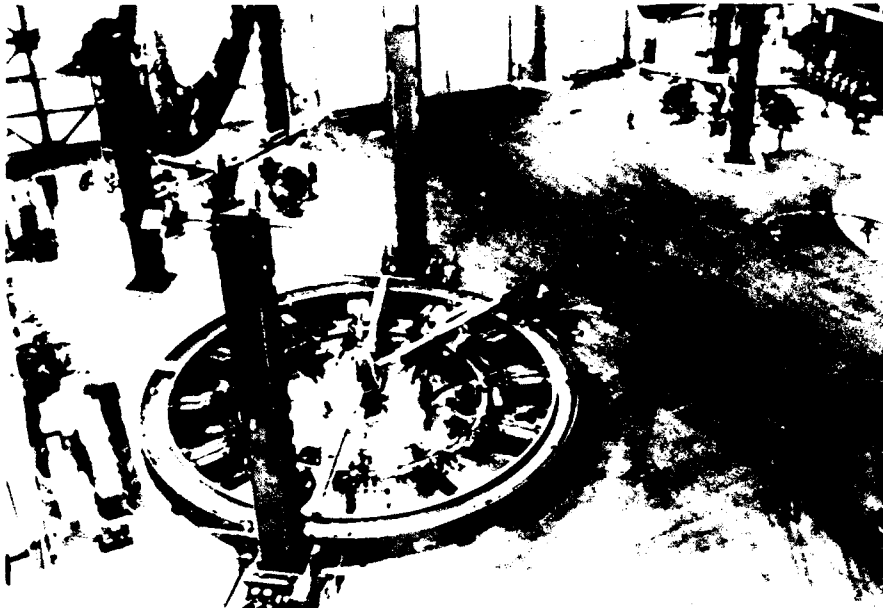
—All tooling must be proved before its incorporation into the manufacturing process. Tooling should be proved on three-dimensional objects.

—Prime contractors must maintain visibility into their subcontractors' tooling control plans.

—Using soft tooling during rate production increases process variability and puts your program at risk.



Fixtures and jiggig used in the construction of submarine hulls.



Fixtures and jiggling used in the construction of submarine hulls.

Related Definitions

—*Jig*; a device for locating, supporting and holding a workpiece in a fixed position while guiding a tool to the workpiece.

—*Fixture*; same as a jig except it cannot guide a tool.

—*Special Tooling*; all hardware, not commercially available, that is required to produce a product.

—*Soft Tooling*; denotes flexibility of use rather than a physical characteristic. Soft tooling typically is used during low-rate production, is cheaper to design and fabricate, can be modified quickly but is not durable or as accurate as hard tooling.

—*Hard Tooling*; denotes inflexibility of use rather than a physical characteristic. Hard tooling typically is used during full-rate production, usually is very expensive to design and fabricate, usually cannot be altered but is very durable and much more accurate than soft tooling.

—*Expandable Rate Tooling*; a tooling concept that tries to combine the flexibility of soft tooling with the accuracy of hard tooling; generally applied to assembly tooling. Basic con-

cept consists of a permanent (hard) external frame combined with temporary (soft) locators that are positioned with theodolites. Frames can, therefore, be used to assemble more than one component.

—*Tool*; a device used for the purpose of removing material from a workpiece under controlled conditions, such as a drill bit.

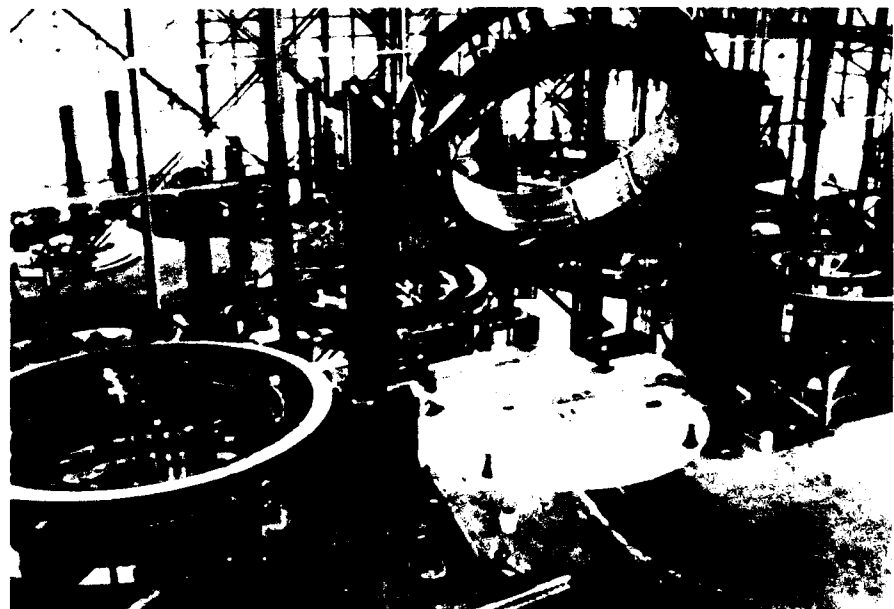
Assembly tooling can be hundreds of feet long and many stories high, depending on how many detail parts are to be assembled together and the size of the finished product.

References

Best Practices, U.S. Navy NAVSO P-6071, Section 6.6

ANSI Y14.5M, "Dimensioning and Tolerancing."

Production Course (draft), U.S. Navy Technical Education Program, Office of the Assistant Secretary of the Navy, (Product Integrity) 1992.



Fixtures and jiggling used in the construction of submarine hulls.

RESULTS-ORIENTED PROGRAM MANAGEMENT AS A LEADERSHIP/ MANAGEMENT MODEL

*For DOD Information Systems
Program Managers*

James E. Price

Mary-Blair Valentine

Results-oriented program management (ROPM) is an attitude of effective program/project managers. It mandates careful stewardship of resources and delivering a product satisfying the customer. The ROM develops executive competencies in senior managers, challenging them to mold organizational culture to emphasize long-term goals and quality, in spite of the daily competing and conflicting operational requirements.

When identifying effectiveness criteria for managing information systems, literature is uniformly consistent. The design, development and fielding of an information system is

Mr. Price is a Professor of Information Systems Management at the Defense Systems Management College.

Ms. Valentine serves as the Army Health Promotion Program Coordinator and Communities of Excellence Program Manager for the Army Materiel Command.

deemed successful when it satisfies user requirements, is produced within budget and completed on time. The order of priority may be changed, but these criteria are universally accepted by developers of information systems.¹ These criteria comprise the foundation of results-oriented program management.

DOD Program Management: Information Systems Development

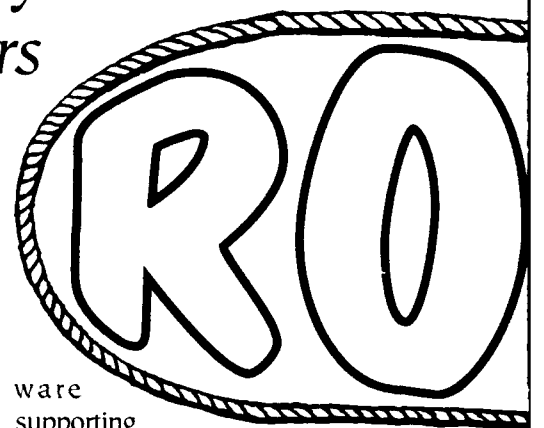
The Department of Defense is a complex environment. Disparate elements must work independently, or in close synchronization, depending on changing circumstances. The overall mission of defending the nation's security is subdivided into Army missions, Navy/Marine goals and Air Force functions. The customer is, alternately, the soldier in the desert, American taxpayers and their families, the Congress, or the global village.

The DOD, spending \$30 billion annually on computer software, is the largest producer of software in the world.² Much goes for embedded soft-

ware supporting high-tech military equipment like airplanes, ships, tanks and communications instruments such as those used in Operation Desert Storm. Of that figure, \$9 billion³ is spent on information system initiatives.⁴ Multimillion dollar systems are common. In fact, the DOD doesn't consider an information system large unless life-cycle costs top \$100 million.

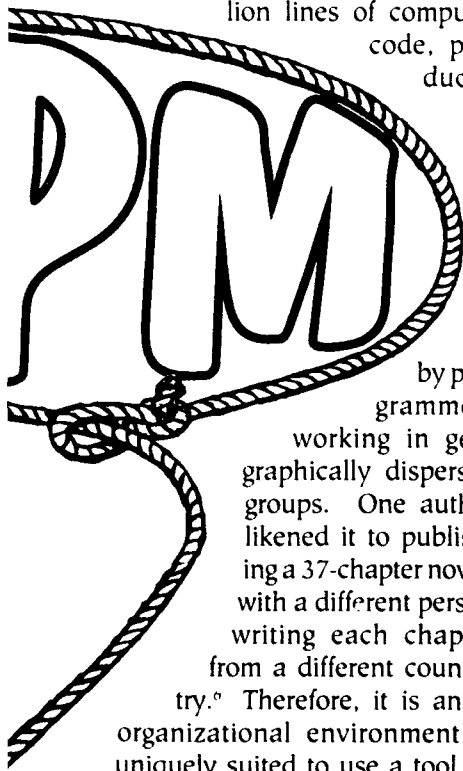
Developing information systems has unique features that make it interesting to investigate. It involves large PM teams and is difficult to measure progress or quality short of completion. It must be done right the first time. Historically, it is plagued with high turnover of personnel and requires careful stewardship of taxpayer dollars.

The program management environment in the Department of Defense is similar to any large bureaucratic



organization; that is, except for sizes of program offices and the cost of information systems. When representatives from user community⁵ organizations are considered, it is not unusual for a DOD program "team" to include hundreds. When private-sector contractors join the team, numbers can be astronomical.

Consider this: One programmer can be expected to write, test and debug 2,000 lines of computer code in 1 year. The number of programmers involved becomes mind-boggling when you consider many DOD information systems contain several million lines of computer code, produced



by programmers working in geographically dispersed groups. One author likened it to publishing a 37-chapter novel, with a different person writing each chapter from a different country.⁶ Therefore, it is an organizational environment uniquely suited to use a tool like ROPM to improve effectiveness.

How to Implement Results-Oriented Program Management

The concept of results-oriented program management keeps program managers focused on the big picture. An attitude component connects this concept with organizational culture, and its goal is to meet customer/constituent needs. It prevents program managers from getting lost in day-to-day details. The problem is that senior management must imbed this results-oriented attitude into organizational culture. This requires seasoned management skills. It raises the question: What, specifically, are attitudes, behaviors and competencies that make a results-oriented program manager?

Profile of Results-Oriented Program Manager—ABCs

An extensive content analysis of effective management criteria, with our empirical research, surfaced attitudes, behaviors, and competencies used by results-oriented program managers.⁷ The criteria provide the ABCs of an effective results-oriented program manager as described below.

—Successful program managers have a sense of ownership in their programs, believe in the mission and can translate that vision to the program

team with written and oral communication, and by example.

—They maintain an efficiency orientation, and develop program-office procedures to ensure resources entrusted to them are used wisely. They organize the program office to optimize workers' contributions and track task accomplishment and milestones toward program completion. They create an environment with a focus on excellence and results-oriented performance.

—Effective managers are oriented to action and are concerned with impact. Looking beyond the program office, the results-oriented program manager is aggressive and assertive in influence within the tridimensional environment. Organization reputation and the prudent use of power are key concerns.

—They use strategic influence and fondness for proactivity to make things happen. With a firm grip on overall program strategy and goals, they continually gather information, introduce change, prevent mishaps, and monitor performance, budget, and timelines for potential glitches. Ultimately, they must accept blame for program failure or share praise for program success—both require hands-on, proactive, effective management.

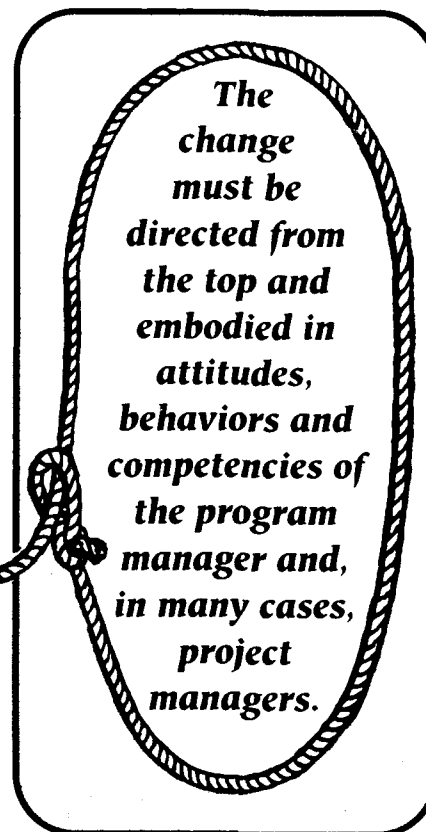


—Results-oriented program managers radiate self-confidence and will take the initiative to ensure program success.

—Successful program managers can conceptualize goals that must be met at the end of the program. Using innovation and creativity, the effective program manager offers insight regarding goals, their development, and ultimate attainment. Using inductive reasoning the program manager analyzes and describes patterns, cycles and relationships. The PM develops concepts and themes to guide managerial processes, transmitting them to employees and stakeholders by appropriate models, metaphors, analogies, etc.

—They use deductive reasoning to diagnose or interpret events occurring along the way to program completion. Systematic thought is required to analyze key situations and keep the program on track. Critical inquiry keeps the program from becoming stagnant or irrelevant to changing needs and fluctuating demands.

—Effective managers possess political awareness and know how to use socialized power wisely. The program office functions in a highly politicized environment in which the effective program manager must establish and maintain operational networks to obtain cooperation and coalescence on program goals. The program manager of a major program has the power



to influence key players in the tridimensional environment and in the program outcome. The PM is a role model for employees, advocate for the program, and negotiator for, and steward of, required resources.

—They are managers of group processes. Major programs cannot be completed successfully without a smoothly functioning team. The program manager develops relationships at all levels of the tridimensional environment, but is particularly concerned with ensuring the program team is selected, molded, guided and rewarded for collaborative, facilitative teamwork and

for performance targeted toward program goal attainment. The PM must instill vision on the big picture so that each employee knows how critical each role is in the program's success. In this way, everyone may feel ownership in the program.

—The results-oriented program manager described is a change agent. Implementation of results-oriented program management requires an attitudinal change focusing everyone on the big picture. Why are we here? The answer is: To produce an information system meeting user requirements, within budget and on time. Short-range activities should be planned and executed with this question in mind: How will action today impact on our long-term goal?

It is as change agents that results-oriented program managers have the greatest impact on the organization. If the organization is to adopt attitudes, behaviors and competencies to meet customer needs, there must be change. The change must be directed from the top and embodied in attitudes, behaviors and competencies of the program manager and, in many cases, project managers.

Results-Oriented Program Management As a Leadership/Management Model

The bottom line is this: If an organization is to change, in this case to infuse itself with an emphasis on quality and an orientation toward results, the existing organization culture must be modified. If employees are to respond, new culture and new vision must be presented in a way that enables adapting personal goals. Employees' attitudes must be molded to personally accept organizational strategic goals to complete an information system meeting user requirements, within budget and on time.

Lewin's change model is an effective way to look at this process. For the new vision to replace current re-

FIGURE 1. Why We Need ROPM

99.9 PERCENT SUPPLIERS PROVIDE:

1. 20,000 incorrectly filled prescriptions annually
2. Unsafe drinking water one hour a month
3. No electricity, water, or heat for 8.6 hours every month
4. 500 incorrectly performed surgical procedures every week
5. 2 long (or short) landings a day at O'Hare (also LAX, NY, Atlanta, etc.)
6. 2,000 pieces of mail lost every hour

Source Unknown

FIGURE 2. Barriers to Change

ATTITUDINAL BARRIERS

We've Never Done It That Way
We're Not Ready For That
We're Doing OK Without It
We've Tried It Before
It Costs Too Much
That's Not Our Job
It's Not Practical

CULTURAL BARRIERS

Lack of Vision
Lack of Understanding/Belief
Environment of Low Risk-Taking

SOCIAL BARRIERS

Communication

ORGANIZATIONAL BARRIERS

Lack of Management Support
Poor Planning

PSYCHOLOGICAL BARRIERS

Track Record of Resistance

ality, old ways must be gradually eliminated (unfrozen) and replaced with the new (changing). Employees must participate in the change process. If properly prepared for upcoming changes with clear rationale, new behaviors will become internalized (refrozen) easier than if sweeping change is mandated from the top with no explanation to, preparation with, or participation from those affected.⁸

Molding Organization Culture

Results-oriented program management encourages an organization-wide attitudinal change, which requires a substantial effort spearheaded by senior management. An organization's culture influences its strategy, structure, operating procedures and interpersonal relationships. Program managers need to mold an organizational culture supporting their views of how

team members should perceive, ponder and resolve problems.

Structural changes must be made within the organization. For example: Lines of authority, span of control, division of labor, organizational policies and procedures, functional department relationships, communications, and control of human resources probably will be modified to create an organizational structure promoting a culture supportive of senior management's long-term vision.

Corporations feel the need to match Japanese quality to stay competitive. The Department of Defense must ensure prudent use of taxpayer dollars. But, there are human concerns for having quality pervade the way we work. Figure 1, lists the service we may expect if settling for 99.9 percent quality. One-tenth of one percent makes a difference!

In the information-systems arena, customer needs are not met if there are too many lines of faulty code or if debugging is a longer and costlier process than the development itself. In the Department of Defense, software errors can mean the difference between a missile hitting its target, or accuracy of high-tech weaponry on which our security relies.

Although no one would argue that 100-percent quality is the benchmark of success, imbuing the organization with the desire to produce quality information systems, on time and within budget and satisfying customer needs, is not easy. There are hurdles to overcome.

Implementation Obstacles in Program Management Environment

Implementing ROPM represents an organizational change, and many people resist change. Why? People are creatures of habit and need to be convinced to change before willingly

doing so. People often are afraid of the unknown and want to retain the status quo. Some barriers to change (Figure 2) that must be overcome before implementing ROPM can be attitudinal, organizational, cultural, social or psychological.

Other implementation obstacles are prevalent in the program management environment (Figure 3). Program management organizations often are in a product-development mode that ends when the objective is met. Assigned and attached personnel may lack loyalty and commitment to goals of the program organization, and may be rewarded for measurable performance without regard to quality of output.

Many changes affecting personnel are concerned with new work settings and groupings. According to Gabor:

...most companies find they must also teach employees who for years have worked in functional fiefdoms how to operate in multidisciplinary teams. Industry desperately needs to foster teamwork. The only training or education on teamwork our people receive in school is on the athletic field. Teamwork in the classroom is called cheating.⁹

FIGURE 3. Program Management Environment

1. Putting out daily fires
2. Short-term focus
3. The battle of the budget
4. Preparing for next major milestone approval
5. Contract management
6. Ever-changing user requirements
7. "Selling" the new approach
8. Day-to-day adventures

| | |
|----------|-----------------|
| Travel | Working lunches |
| Meetings | Long hours |
| Calls | In-basket |

This is apropos in the matrix environment of program and project management. The challenge of blending assigned and attached personnel into a smoothly functioning team, in concert with political and other external players, is a key role for the results-oriented program manager. In geographically dispersed organizations, it becomes more critical that every player in every location ascribe to the same attitudes, behaviors and competencies to ensure customer needs are met.

ROPM: A Model for Change

Program management offices for DOD information systems need results-oriented program managers. The PM offices are not insulated from the need to shift attitudes, behaviors and competencies to those who encourage proactive change, effectiveness and efficiency. To do this, the PM officers must instill a vision of success and a willingness to forego the status quo. Rosabeth Moss Kanter may have expressed it best in her book, *The Change Masters*.

...the more profound, comprehensive, and widespread the proposed change, the more absolute is the need for deep understanding and active leadership by the top managers....¹⁰

Complexity and size of DOD information systems organizations make them prime targets for the kinds of results offered by using results-oriented program management. The matrix structure of the project office is a perfect environment for ROPM team-building.

It is extremely difficult for program managers to keep their spotlights on long-term goals with competing, and often conflicting, operational requirements. Nevertheless, PMs must move beyond the process orientation often found in bureaucratic organizations and focus on producing results. The question is: Where do they look?

If the management style of organization leaders reflects a commitment to results-oriented program management, all personnel with that perspective will perpetuate the attitude. An organization open to change will have no difficulty in continually improving processes for the betterment of suppliers and customers.

Results-oriented program management's emphasis on the big picture can only enhance inevitability of program success. In the program-management environment, results-oriented program management is a tool suitable for beginning the process of redirecting organizational focus to quality; that is, developing an information system (or any product or service) that meets customer needs, within budget, and on time.

The results-oriented program manager is the change agent guiding the organization to this new way of doing business, and to greater success in meeting organizational goals.

Endnotes

1. Atkinson, Rick and Barton Gellman, "Navy Fires Three Working on Top Jet: Lagging Schedule, Cost of A-12 Cited," *The Washington Post*, December 5, 1990, pp. 1 and 19; Bobrowski, Paul M., "Project Management Control Problems: An Information Systems Focus," *Project Management Journal*, 1989, p. 11; Kerzner, Harold, "In Search of Excellence in Project Management," *Journal of Systems Management*, February 2, 1987, p. 31; Moir, James, "Project Methodology, Organization, and Structure," *Journal of Information Management*, p. 38; Pinto, Jeffrey K. and Dennis P. Slevin, "Critical Success Factors in Effective Project Implementation," ed. Cleland, David I. and King, William R., *Project Management Handbook*, New York: Van Nostrand Reinhold, Second Edition, 1988, pp. 481-482;

and Thayer, Richard H., Arthur B. Pyster, and Robert C. Wood, "Major Issues in Software Engineering Project Management," *IEEE Transactions on Software Engineering*, Vol. SE7, No. 4, July 1981, p. 333.

2. Richards, Evelyn, "Pentagon Finds High-Tech Projects Hard to Manage," *The Washington Post*, December 11, 1990, p. 1.

3. Department of Defense, Memorandum from Donald J. Atwood, Deputy Secretary of Defense, to senior DOD officials, dated October 4, 1989, Subject, DOD Corporate Information Management.

4. Information System is synonymous with Management Information System (MIS).

5. Functional members (acquisition experts, contract managers, financial experts, technical experts, etc.) called upon to provide assistance.

6. Marsh, Alton, "Pentagon Up Against a Software Wall," *Government Executive*, Vol. 22, No. 6, May 1990, p. 63.

7. Boyatzis, Richard E., *The Competent Manager: A Model for Effective Performance*, New York: John Wiley, 1982; Department of Defense, *A Competency Model of Program Managers in the DOD Acquisition Process*, Fort Belvoir, Va., Defense Systems Management College, 1989; and, our empirical research.

8. Lewin, Kurt, *Field Theory in Social Science*, New York, Harper and Row, 1951.

9. Gabor, Andrea, *The Man Who Discovered Quality: How W. Edwards Deming Brought the Quality Revolution to America*, New York, Times Books/Random House, 1990.

10. Kanter, Rosabeth Moss, *The Change Masters: Innovation for Productivity in the American Corporation*, New York, Simon and Schuster, 1983.

AN UNTRAINED PROGRAM MANAGER CAN HAVE MONUMENTAL RESULTS



Program managers can't be built like a tank or a plane. It takes a solid education to be successful. The Defense Systems Management College welcomes industry executives, government civilians and military officers. Attend one of DSMC's 22 short courses and workshops—many of them ACP mandatory—at any of our five locations around the country. Or enroll in the 20-week Program Management Course at our main Fort Belvoir campus. For a college catalog, call (703) 805-2227/2850 or DSN 635-2227/2850, or write DSMC (OS PS 87), FORT BELVOIR, VA 22060-8426.

ORGANIZATION DEVELOPMENT

What Is It? How Can You Use It?

Major Jim Wilson, USA

One fact about the business climate stands out from all others. The pace and effect of change is greater than ever before. New products, new methods of doing business and new competitors pose problems that have no precedent. In order to meet today's challenges it is essential that organizations use their resources in the most effective ways possible. The organizational development (OD) process and skilled OD practitioners are valuable tools in the effort to remain competitive and profitable.

The past 30 to 35 years have seen evolution of a new service and profession which is called organization development. This article provides a general overview of this profession and presents ideas on how the service can be used in your program management or acquisition organization to improve

your products or services. After a general description and background of OD, I will "walk through" a typical OD "intervention."

Organizational development is a systematic effort to improve an organization based on melding of organization theory with behavioral theory. It depends on an effective process of

gathering information, interpreting that information and making plans from it, implementing those plans and evaluating the implementation to

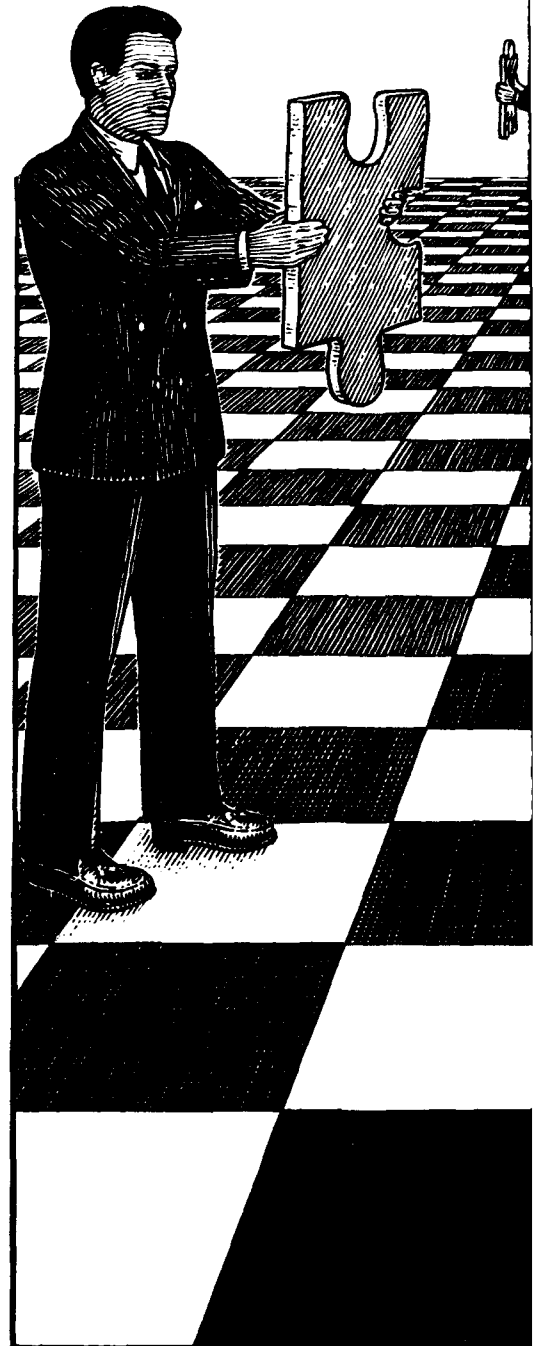
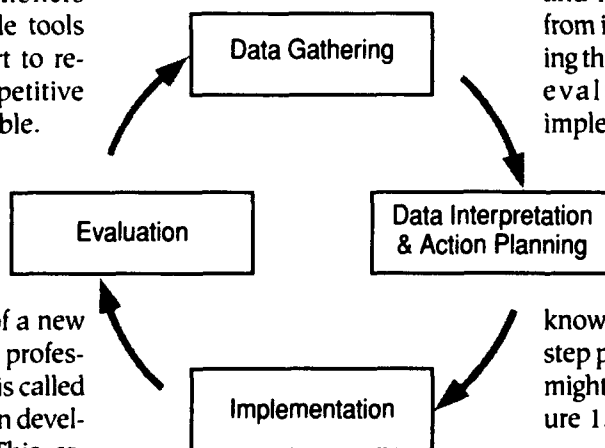
determine effectiveness. This is sometimes

known as the four-step process which might look like Figure 1.

These four steps should be considered as a never-ending cycle because, as long as the organization exists, there are changes happening that need to be managed effectively. The OD provides a method for using the process of change to the organization's benefit.

Organizations can be seen as having two parts, a social system and a

FIGURE 1. Four-Step Process



Major Wilson is a Professor in the Managerial Development Department at the Defense Systems Management College.

technical system. The technical system is the content of the business and is peculiar to each type of business. In the building industry, for example, the technical system revolves around the knowledge skills and abilities to provide technical plans, pour concrete, bend metal, cut wood, etc. The social system of an organization is the means by which people interact to successfully implement the technical system. The OD improves organizations through collaborative processes that focus on the social system of the organization as the means

of improving the social and technical systems.

A helpful way to examine the organization is through the use of systems models. I will discuss an example of a systems model later.

A second element of the method is to understand the assumptions underlying OD. Some of the most important are:

—Groups of people are the basic building blocks of organizations

so working with groups is the primary way of effecting change.

—Honest and supportive relationships are critical to achieving a high-performing organization.

—People seek and are most effective in situations where they have an opportunity to contribute responsibly to achieving goals and are recognized and rewarded for doing so.

—Collaboration and participation are the most effective ways to implement change and improve effectiveness.

The third element of the OD method mentioned here is the use of a trained OD consultant or practitioner. There are several roles that consultants typically take in organizations: The expert role, where the expert provides direction about what to do; the staff role, where the consultant is seen to be an extension of the existing staff; and the collaborative or facilitative role where the consultant gathers information and helps the organization focus on issues it decides are important. The OD consultant functions in the collaborative or facilitative role. This will be made clearer when we "walk through" a typical OD "intervention."

I mentioned one of the ways to examine the organization is through the use of systems models, which serves as a framework upon which data about the client company can be structured. This allows systematic study that helps to understand the complex functioning of the system. I will describe one widely used model, the "six box" and use it in our "walk-through".¹

In the six-box model, each box represents a subsystem of the organization. They are: purpose, structure, leadership, relationships, rewards and helpful mechanisms boxes. Information gathered about the organization in the data gathering or assessment

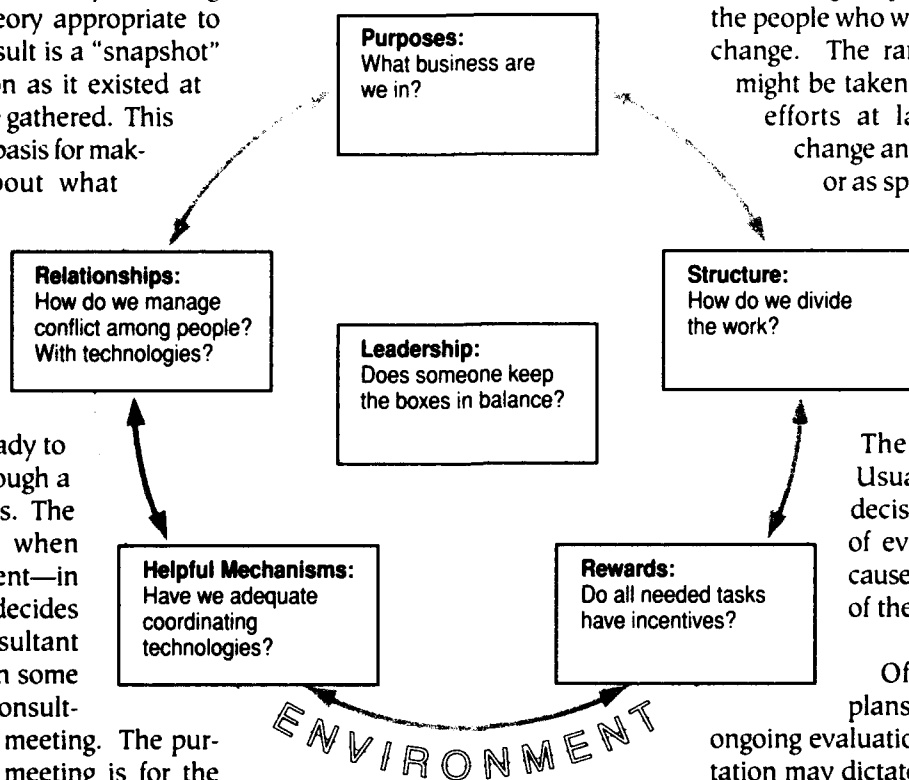


phase of OD interventions will fit into one or more of these boxes. This information is then analyzed using organizational theory appropriate to each box. The result is a "snapshot" of the organization as it existed at the time data were gathered. This information is the basis for making decisions about what change is desirable in the organization. This model could look like Figure 2.

We are now ready to begin walking through a typical OD process. The first step occurs when someone—the client—in an organization decides that an OD consultant might be of help in some way, contacts a consultant and sets up a meeting. The purpose of this first meeting is for the prospective client to lay out the concerns that motivated calling a consultant. The consultant can be expected to ask clarifying questions to help focus the prospective client on current concerns. At some point, a tentative decision is made by both parties regarding desirability of working together to solve whatever problems may exist.

A prospective client could expect several meetings with the consultant to refine specifics and negotiate a contract that spells out the specifics of what is to be done. Some issues that might be mentioned are: objectives of the project, means of gathering information, who the client is, kind of information sought, any boundaries on information sought, each person's role in the project, product to be delivered by the consultant, support and involvement expected of the client, a time schedule, confidentiality and anonymity of information, feedback of data to the client and other participants in the project, fees and payment schedules or procedures for ter-

FIGURE 2. Systems Model



mination, and any restrictions placed on the consultant.

Once client and consultant reach an agreement and sign a contract the consultant will implement the four-step process. Information will be gathered by one or more of several methods such as interviews, surveys, and study of company records. The data gathering objectives depend on objectives of the project. The data gathering can be broad, covering many aspects of the system, or more narrowly focused on a specific system part. The information is organized according to some method such as the six-box systems model and analyzed by the consultant. In some cases, the client and consultant may have decided to jointly analyze the data. In any case, the information is presented to the client in a feedback meeting.

Based on results of the feedback meeting some decisions may be made regarding issues to be addressed or

problems to be solved. Action plans are created by the people responsible for effecting the proposed change, and the people who will be affected by the change. The range of actions that might be taken can be as broad as efforts at large-scale cultural change and strategic planning, or as specific as team building with one or more work teams. (The consultant is not responsible for implementing the changes. The organization is.) Usually, there will be a decision on the method of evaluating the effects caused by implementation of the action plans.

Often, when action plans are implemented, ongoing evaluation of the implementation may dictate midstream corrections. Final evaluation of the effects of the implementation of action plans will be made as the implementation is finished. This data can then serve as a basis for further data gathering and future actions. The four-step process provides a method for continuous process improvement, a fundamental aspect of total quality management.

The organization development process and OD consultants are valuable tools in the effort to become more competitive and profitable. Leaders of organizations can use the process to accomplish many goals. To learn more about the process feel free to call Major Jim Wilson at the Defense Systems Management College, commercial (703) 805-3425.

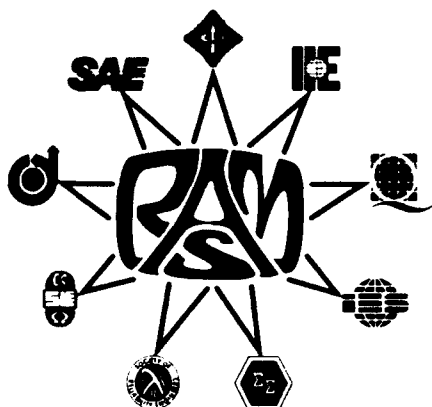
Endnotes

1. The six-box model was created by Marvin Weisboard.

1993 ANNUAL RELIABILITY AND MAINTAINABILITY SYMPOSIUM

Westin Peachtree Plaza Hotel
Atlanta, GA USA

January 25-28, 1993



THEME ASSURANCE TECHNOLOGIES FOR TOMORROW'S ENVIRONMENT

- Total Quality Management
- R&M Management
- Software R&M
- R&M Case Studies
- Maintainability/Logistics
- Reliability Growth/Testing
- Commercial Reliability
- R&M Tutorials
- International Standards
- R&M with Decreasing Budgets

For More Information Contact:

R.G. Schueppert, Jr.
Beckman Instruments, Inc.
200 S. Kraemer Blvd.
MS: W351
Brea, CA 92621 USA
(714)961-3793
FAX (714)961-3743

SPONSORING SOCIETIES



STATEMENT REQUIRED BY THE ACT OF AUGUST 12, 1970, SECTION 3685, TITLE 39, UNITED STATES CODE. SHOWING THE OWNERSHIP, MANAGEMENT, AND CIRCULATION OF

Program Manager, published bimonthly at the Defense Systems Management College, Fort Belvoir, VA 22060-5426. Number of issues published annually: 6. The Director of Publications is Robert W. Ball, Defense Systems Management College, RD-P, Fort Belvoir, VA 22060-5426. The Managing Editor is Catherine M. Clark, Defense Systems Management College, RD-P, Fort Belvoir, VA 22060-5426. The publisher is the Defense Systems Management College, Fort Belvoir, VA 22060-5426.

The average number of copies each issue during the preceding 12 months:

- A. Total number of copies printed (net press run): 13,083
- B. Paid and/or requested circulation: 1,500
 - 1. Sales through dealers and carriers, street vendors, and counter sales: None
 - 2. Mail subscriptions paid and/or requested: 10,000
- C. Total paid and/or requested circulation: 11,500
- D. Free distribution by mail, carrier, or other means, samples, complimentary, and other free copies: 1,500
- E. Total distribution: 13,000
- F. Copies not distributed
 - 1. Office use, leftover, unaccounted, spoiled after printing: 83
 - 2. Returns from news agents: None
- G. Total distribution: 13,083

The actual number copies of single issue published nearest to filing date:

- A. Total number of copies printed (net press run): 10,500
- B. Paid and/or requested circulation: 1,500
 - 1. Sales through dealers and carriers, street vendors, and counter sales: None
 - 2. Mail subscriptions paid and/or requested: 7,000
- C. Total paid and/or requested circulation: 8,500
- D. Free distribution by mail, carrier, or other means, samples, complimentary, and other free copies: 1,500
- E. Total distribution: 10,000
- F. Copies not distributed:
 - 1. Office use, leftover, unaccounted, spoiled after printing: 500
 - 2. Return from news agents: None
- G. Total distribution: 10,500

THE SOFTWARE DEVELOPMENT PROCESS

An Integral Part of Weapon Systems

Captain Jeanne L. Murtagh, USAF

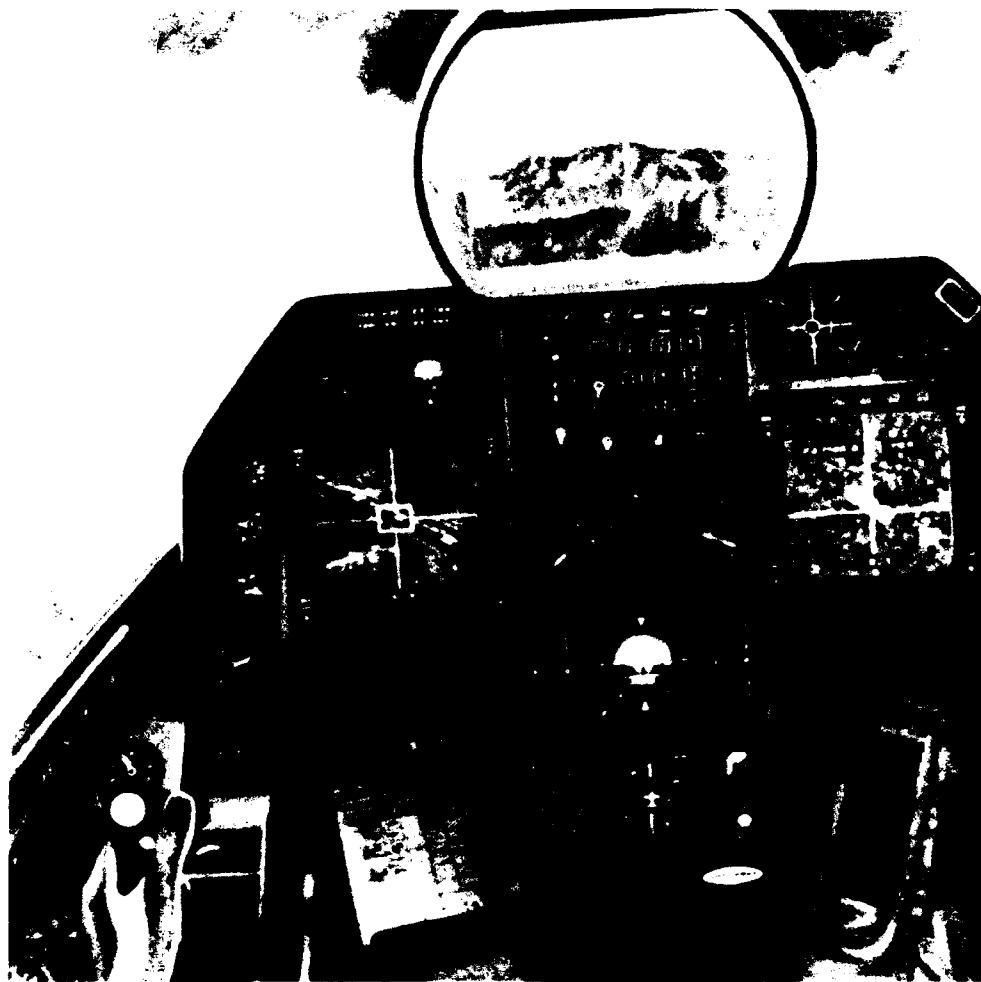
Software is an integral part of virtually all weapon systems, so it's useful for anyone involved in weapon-systems development to understand the software development process. This article covers the definition of mission critical computer resources and addresses the process used to develop this type of software.

Definition of Mission Critical Computer Resources

What are mission critical computer resources (MCCR)? This includes hardware and software critical to a military application. These resources are usually "embedded" in another system. The software in MCCR tends to be harder to develop than software for other types of applications. This is primarily because most MCCR software must operate in "real-time." In other words, there are very rigid time constraints on this type of software. This may become clearer with an example.

The fire control system on a fighter aircraft contains MCCR. The purpose of the hardware and software embedded in this fire control system is to accept information from the sen-

Captain Murtagh is an Instructor, Acquisition Course, 3440 Technical Training Squadron, Lowry Air Force Base, Colorado. She has 8 years of acquisition experience and holds an M.S. degree in computer science.



The purpose of the hardware and software embedded in the fire control system on a fighter aircraft is to accept information from the sensors, identify targets, and select and launch weapons against these targets.

sors on the aircraft, identify targets, and select and launch weapons against these targets. Software in the fire control system must accept a message from the fighter's radar stating the identity and position of the target. If much time elapsed between receipt

of this message, and selection and launch of a weapon, the target's position would probably have changed enough to cause the weapon to miss it. The amount of time which could be allowed to elapse without causing difficulty would probably be measured

in milliseconds. Contrast this with a typical data processing example. It really should not matter if it takes an extra four minutes to run this month's payroll.

The development of MCCR software is governed by DOD-STD-2167A. This standard describes steps in the software development process, and the documentation and reviews and audits associated with each step. The standard can be tailored to accommodate each program's needs.

Approach to Software Development

Before we can discuss details of the software development process, we need to address more general issues. Designing large computer programs is a difficult process. It requires us to manage huge amounts of complicated information, and we humans have a limited capacity to do that. We'll look at the research basis for that limited capacity first. Then we'll talk about an approach to software design which helps us organize all the information we need so that we can manage it. This approach involves

dividing our computer program into pieces small enough to understand. We'll talk about specific names for each piece of the computer program. This will provide you with enough information to understand details of the software design process.

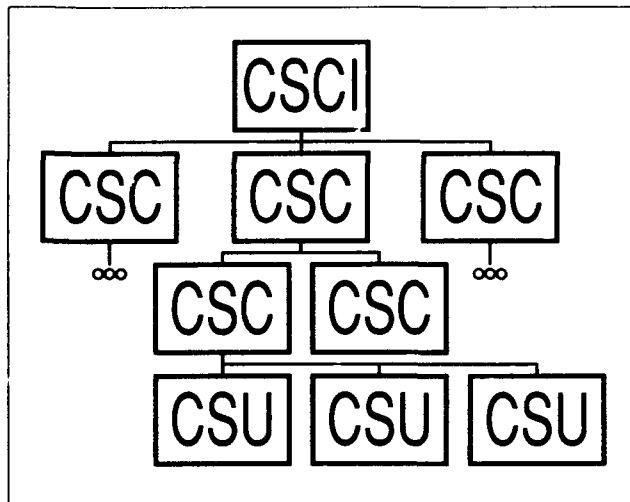
"Seven, Plus Or Minus Two, Rule"

Humans have a limited capability to manage unrelated information. We normally only remember and work with seven, plus or minus two, items or issues at a time. When the number of items you need to remember and work with—at one time—exceeds this limit, you need to start "chunking" (or grouping) items into related categories. The seven, plus or minus two, rule is based on research done by George Miller at Harvard University. Additional information can be found in *The Brain Book*, by Peter Russell.

This restriction on capacity to manage unrelated facts explains why we cannot attack the development of a huge military computer program in one step. Most military computer programs (for example, the software that drives the flight control surfaces in the "fly-by-wire" F-16) are very large. The minimum number of lines of code (or individual instructions) in this type of program is usually around 30,000. If we tried to tackle that project in just one step, we would clearly exceed our capacity to keep track of all pieces of information (the lines of code) and how they should work together.

We need to somehow divide our problem—how to develop the computer

FIGURE 1. Anatomy of a CSCI



program—into smaller, related pieces, so we won't exceed our capacity to manage the parts of the problem.

Top-Down Design Process

The top-down design process lets us do just that. The basic idea is that we start with a very broad definition of what the computer program must do, and then progressively refine this definition, adding more details at each step. This very broad definition is abstract; it does not contain detailed information. As we add details at each step, our solution becomes less abstract and more concrete; it becomes progressively more specific. This lets us break a large problem into several smaller problems. Then we apply this same approach to each of the smaller problems, until they get small enough that we can retain enough

FIGURE 2. An Analogy

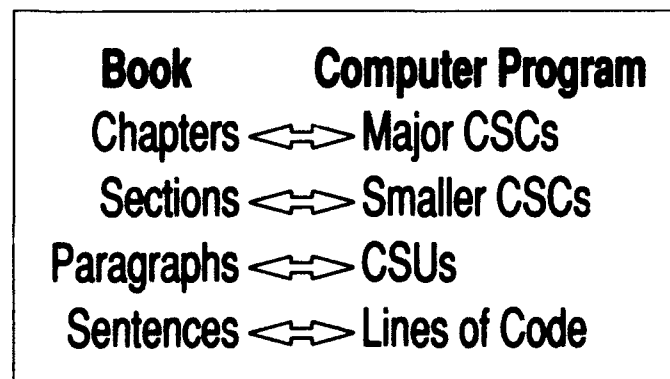
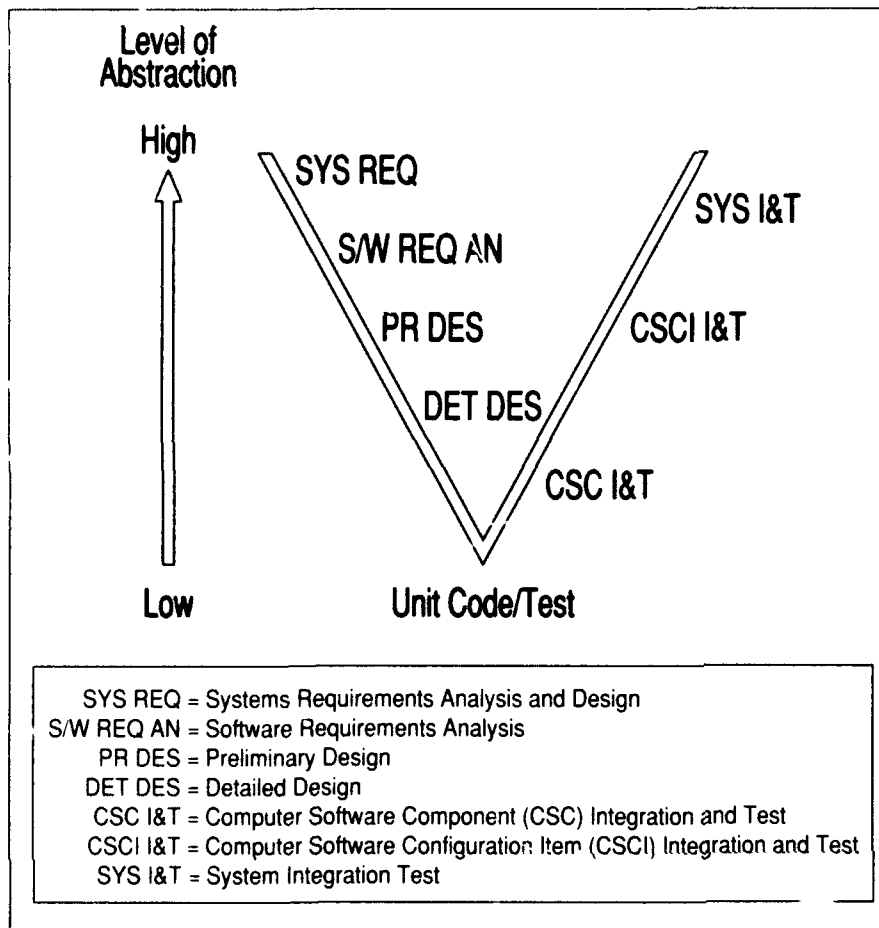


FIGURE 3. Steps in the Software Development Process



information to solve them. This prevents us from violating the seven, plus or minus two, rule, since we need to work on only one of the smaller problems at a time.

Anatomy of a CSCI

Top-down design requires us to divide a computer program into progressively smaller and simpler parts as we design the program. We have specific names for each of these parts (Figure 1).

Each entire computer program is called a computer software configuration item or CSCI.

Each CSCI is divided into a number of computer software components or CSCs. It is common to have several levels of CSCs in the CSCI.

The smallest entity in a CSCI is called a computer software unit or CSU. Programmers have many names for these entities: subprograms, functions, procedures or modules. We often require that no CSU contain more than 100 lines of code (or individual instructions). This helps ensure that the CSU does not get too complex to understand.

You can draw an analogy between writing a large computer program and writing a book. Organizing the entire computer software configuration item is similar to organizing the book. You would break the book into chapters; the CSCI would be broken into its major computer software components. The chapters in the book might be divided into different sections; the major CSCs would each be divided into smaller CSCs. Each section of

the book would be divided into paragraphs; the lowest level CSCs would be divided into CSUs. Paragraphs comprise sentences, which represent one complete thought; CSUs comprise lines of code (LOC), which contain one complete instruction to the computer (Figure 2).

Now that you have a general understanding of the software development process, let's move on to more specific information.

The Software Development Process

This section covers steps in the software development process. I'll discuss what specifications are developed during each step, and talk about software reviews and audits that occur.

Figure 3 places steps in the software development process on a software development V (See Figure 1). This "V" should help you see how systems are designed and built.

Notice that the software development process starts at a high level of abstraction, progresses to a very-low level of abstraction, and then moves back up the "abstraction ladder" during testing. This is how we use top-down design. This helps us manage the complexity of a software development effort. We discussed why this is important in the last section.

Notice, also, that the software "V" emphasizes that software development involves much more than just writing the code!

Let's briefly discuss what happens in each step of the software development process.

System requirements analysis and design is the first step in the software development process. The software team at the contractor's site analyzes the system specification to determine whether software requirements are

complete (for that level of abstraction) and consistent. Results of this analysis are presented at the System Requirements Review (SRR). The software team determines how many computer software configuration items will be required for the system, and allocates the system's software requirements to each of these CSCIs. This will be presented to the government at the System Design Review (SDR). Preliminary software test planning also is started.

Software requirements analysis is the next step. During this step you determine exactly what each CSCI will do. The contractor's software team completes engineering requirements for each CSCI and documents these in the software requirements specifications (SRS). A Software Specification Review (SSR) is held to review the SRS. The contractor's software team analyzes requirements for each CSCI external interface. These requirements are documented in the interface requirements specifications (IRS). A complete list of qualification requirements for each CSCI is prepared during this step. This is included in the SRS for each CSCI.

It is critical that the software requirements analysis step be done well! If you don't know what you are trying to build, it will be impossible to determine how to build it. We sometimes are tempted to accept software requirements specifications that are not adequate in order to stay on schedule. We must guard against this. Software requirements specifications are the foundation for the entire software development effort. If we accept incomplete or inconsistent requirements specifications, we will maintain the on-schedule illusion (since documents will be authenticated on time), but we will not actually be on schedule (since there are problems that we will have to manage sometime)! Eventually, we will hit a step in the software development process (for example, CSCI testing) which requires us to actually make the software work. At

this point, we have to fix problems. Here's the catch: We now have more than our original problem to manage. Those requirements errors have been propagated into the designs, source code and test plans. We will have to fix these, in addition to the original problem. Thus, the result of failing to admit that the schedule needed to slip a small amount in order to ensure that requirements were adequate is a much larger schedule slip later in the program.

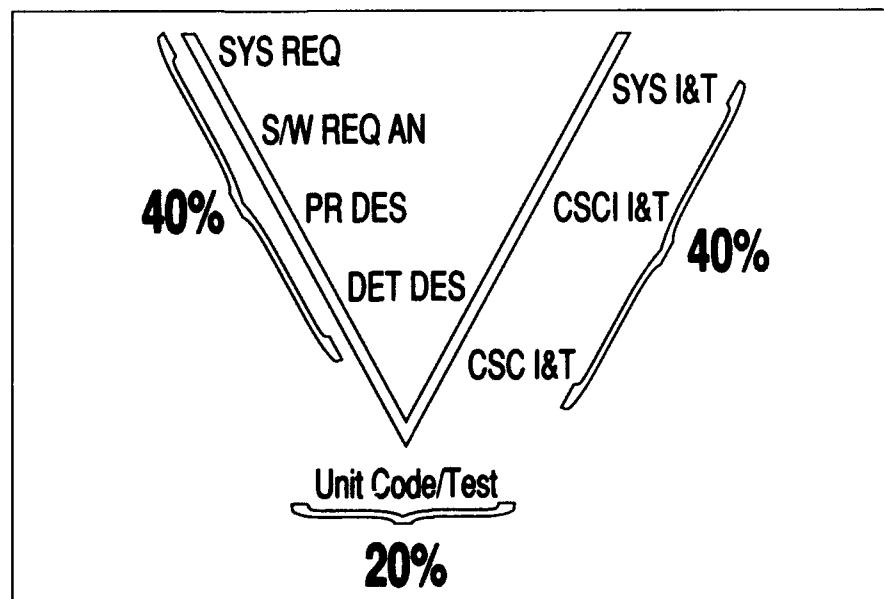
Preliminary design is the next step. You have determined what you want to build; now you must determine how to build it. The contractor's software team develops a preliminary design for each CSCI. This is accomplished by decomposing the CSCI into computer software components, and determining what requirements each CSC will incorporate. This preliminary design is documented in the software design document (SDD) for each CSCI. The preliminary design for each CSCI's external interface is developed in this step. This is documented in the interface design document (IDD) for each CSCI interface. An important part of this is ensuring that all requirements from the SRS and the IRS are addressed in this design. This is called traceability of requirements.

Formal qualification tests for this CSCI are also defined now. These are documented in the software test plan (STP). The preliminary design is reviewed at a software preliminary design review (PDR).

Detailed design follows preliminary design. The detailed design for each CSCI is developed now. This is accomplished by allocating requirements from each CSC to the computer software units which will comprise that CSC, and by establishing the design requirements for each CSU. The software design document is expanded to include this information. Requirements from the IRS also are refined into detailed design information. This is added to each interface design document. Formal qualification test planning continues. Specific test cases are developed, based on the information in the software test plan, and this information is documented in the software test description (STD). The software Critical Design Review is conducted. This will result in approval to start writing code.

Code development and unit test is the next step. The contractor drafts the software product specification (SPS) and writes the code for each CSU. The code normally will be reviewed

FIGURE 4. The Software "V"



Software Acronym List

MCCR Mission Critical Computer Resources

Software Development Documents (in chronological order)

SSS System/Segment Specification

SRS Software Requirements Specification

IRS Interface Requirements Specification

SDD Software Design Document
IDD Interface Design Document

SPS Software Product Specification

Other Software Documents (separated into plans and test documentation)

SDP Software Development Plan
CRLCMP Computer Resources Life-cycle Management Plan

STP Software Test Plan
STD Software Test Description
STR Software Test Report

Reviews and Audits (in chronological order)

SRR System Requirements Review

SDR System Design Review

SSR Software Specification Review

PDR Preliminary Design Review
CDR Critical Design Review

TRR Test Readiness Review

FCA Functional Configuration Audit

PCA Physical Configuration Audit

Parts of a Computer Program

CSCI Computer Software Configuration Item

CSC Computer Software Component

CSU Computer Software Unit

at a code walkthrough before it is entered into the computer. During this step, the code is tested at the CSU level. This testing is relatively informal. Plans and procedures for this testing and test results will be documented in the software development files (SDFs). The contractor is now working at the lowest level of abstraction. As each unit passes its unit testing, it is made available for CSC integration and test.

We now have finished moving down the abstraction ladder and we are ready to move up that ladder, assembling our pieces back into an entire system. Computer software component integration and test is the first step in this process. The SDFs, introduced in the preceding paragraph, will be used to document and control testing during this step. Computer software units are assembled into CSCs and tested at that level. Interface problems between the individual CSUs, or between various CSCs, are discovered now. When the test team is convinced that a CSC is functioning correctly, the CSC is made available for the next level of testing. The newly qualified CSC will be integrated with other CSCs at its level, so that a higher-level CSC can be tested. When each CSC at the top level has been tested, these CSCs will be integrated for CSCI testing.

Computer software configuration item integration and test is similar to CSC integration and test. We assemble the CSCs which comprise the CSCI and test the entire computer program. One or more Test Readiness Reviews (TRRs) will be conducted prior to and during this step. This test phase culminates in formal qualification testing. This testing is conducted in accordance with the software test plan and the software test description. Results of this testing are documented in the software test report (STR). A software functional configuration audit (FCA) can be conducted at any time after CSCI testing is successfully completed. This audit

verifies that all requirements have been incorporated into the design, and that these requirements have been adequately tested. The software product specification, which describes how each computer software unit functions, is finalized at the conclusion of this step. A software physical configuration audit (PCA) is conducted to verify that the CSUs match their descriptions in the SPS. This audit might be delayed until successful completion of system test.

The final step is system integration and test. At this point, the hardware and software for the system are integrated and final testing is performed.

The 40-20-40 Rule

Now that you're familiar with each step in the software development process, let's examine one more aspect of the software development "V." Have you ever heard of the 40-20-40 Rule for software development? This rule appears in *Software Engineering: A Practitioner's Approach* by Roger Pressman. It is a general scheduling rule stating that approximately 40 percent of the software development effort will be spent on front-end requirements analysis and design tasks, 20 percent on coding, and a full 40 percent on testing that code.

Note how this relates to the software "V" (See Figure 4). The left side of the "V" represents the first 40 percent of the effort. The bottom of the "V" represents the 20 percent of the effort spent on coding. The right side of the "V" shows the 40 percent of the total effort which is invested in testing.

Software Planning Documents

We have covered what must be done during a software development effort. Let's take a look at the two master-planning documents used to control this process:

- Software Development Plan (SDP)
- Computer Resources Life-cycle Management Plan (CRLCMP)

The SDP is written by the contractor. It tells how that contractor plans to manage the entire software development effort. It contains quite a bit of information that is not program-specific; a particular contractor's approach to software management usually does not change significantly from project to project. The SDP contains information that is specific to your program. The SDP addresses how the contractor will assess and manage risk on your program, and lists highest priority risks. The SDP identifies the resources (people and equipment) the contractor plans to use for your program.

The CRLCMP is written by the government. The Computer Resources Working Group (CRWG) is responsible for this document. It contains the government's plan for managing the computer system throughout its entire life cycle. One of the most important sections addresses how software will be maintained once the system has been fielded. This often requires many long lead items (items that cannot be obtained quickly after an order is placed), like new facilities or computer equipment at the intended maintenance site. More information on the CRLCMP can be found in AFR 800-14.

Conclusion

We've looked at a definition of mission critical computer resources, and at the process used for developing this type of software. We examined the steps in the software development process, and the documentation, reviews and audits associated with each step. We discussed the 40-20-40 Rule, which provides guidance on scheduling a software development effort. We discussed the two key software planning documents.

This information should help you understand the software development process and its impact on your project.

Bibliography

1. Brooks, Frederick P., Jr. *The Mythical Man-Month*, Addison-Wesley Publishing Company, 1975.
2. Department of Defense, *Defense System Software Development*, DoD-

STD-2167A, Government Printing Office, June 4, 1985.

3. Pressman, Roger S. *Software Engineering: A Practitioner's Approach*, McGraw-Hill Publishing Company, 1982.
4. Russell, Peter. *The Brain Book*, Penguin Books USA Incorporated, 1979.



LESSONS LEARNED IN MANUFACTURING MODERNIZATION

Best Practices and Procedures

*Dr. Robert E. Schafrik
Dr. Thomas D. Fiorino*

Air Force acquisition and manufacturing managers have had extensive experience in encouraging the industrial base to modernize its facilities. The goal of these modernization efforts has typically included increased production efficiency, improved product quality and durability, lower product life-cycle costs, and reduced lead times. In retrospect, some modernization efforts were more effective than others.

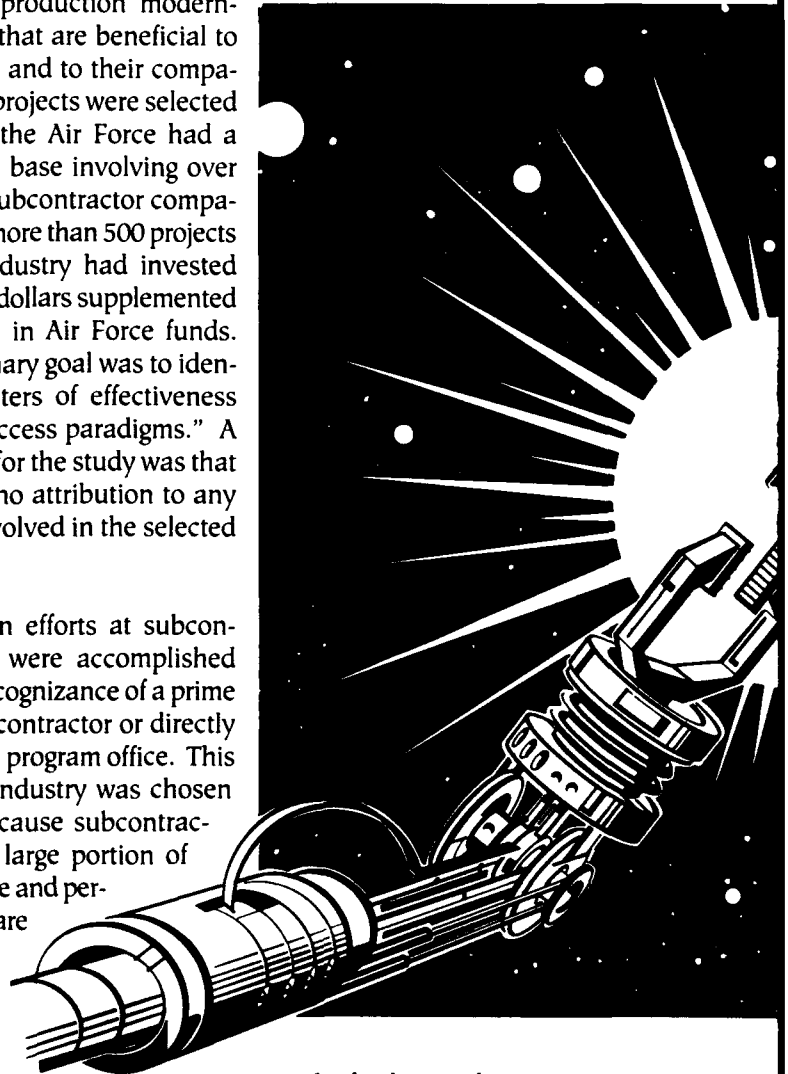
This paper summarizes the key findings of a study that reviewed, analyzed and identified the best practices and procedures of completed facility modernization efforts conducted under the Air Force IMIP program by subcontractors and suppliers.¹ The Department of Defense (DoD) Industrial Modernization Incentives Program (IMIP) was designed to motivate defense contractors to make

investments in production modernization projects that are beneficial to the government, and to their companies. The IMIP projects were selected for study since the Air Force had a large experience base involving over 150 prime and subcontractor companies attempting more than 500 projects in which the industry had invested over \$1.3 billion dollars supplemented by \$560 million in Air Force funds. The study's primary goal was to identify key parameters of effectiveness and provide "success paradigms." A key ground rule for the study was that there would be no attribution to any of the parties involved in the selected projects.

Modernization efforts at subcontractor facilities were accomplished either under the cognizance of a prime weapon system contractor or directly with an Air Force program office. This segment of the industry was chosen for the study because subcontractors comprise a large portion of the industrial base and perform the lion's share of hands-on manufacturing for most major weapon systems.² Many support several weapon systems so that substantial leverage is achieved when their production capabilities are upgraded.

Doctor Schafrik, formerly with Technology Assessment and Transfer, Inc., Annapolis, Md., is associated with the National Research Council's National Materials Advisory Board, Washington, D.C.

Dr. Fiorino, formerly with Technology Assessment and Transfer, Inc., is associated with Production Technology Inc.



The facility modernization process traditionally has comprised four phases.^{3,4} During a company-initiated Phase 0, the contractor and prime/

government establish the overall scope of the effort and define specific goals.

The next phase (Phase I), "Facility Analysis and Project Identification," develops the discipline and blueprint for future factory modernization. The baseline (or "as is") facility is described using a systems engineering approach such as the Integrated Computer Aided Manufacturing Definition Language 0 (IDEF₀).^{5,6} Potential improvement projects are then analyzed to determine the effect on the total facility, and to estimate their development and implementation cost. These projects form the basis for the improved ("to

The goal of these modernization efforts has typically included increased production efficiency, improved product quality and durability, lower product life-cycle costs, and reduced lead times.

potential, improvement concepts, cost/benefit analysis, business strategy related to modernization, capital investment plans, and an implementation time table.

The conceptual facility layouts, developed during Phase I, scope and focus the activity for Phase II, "Project Design and Development." The ma-

ing detailed cost/benefits analyses, refining business strategy, and planning for implementation.

Implementation usually occurs during Phase III, which can be concurrent with part of Phase II. Capital equipment must be purchased and integrated into the company's manufacturing operations, and benefits validated.

The following sections of this paper describe the method used to select the projects for study, the determination of the key characteristics of the successful projects, and key lessons learned for future facility modernization projects.

Selection of Subcontractor/Supplier Projects

A semiquantitative selection methodology was used to rank completed facility modernization projects in order of effectiveness to ensure that programs across the Air Force Material Command were considered on an equal basis. The five criteria used are shown in Table 1. Weight factors and rating guidelines for each criterion were developed based on input from the Air Force program managers and manufacturing experts. This allowed for

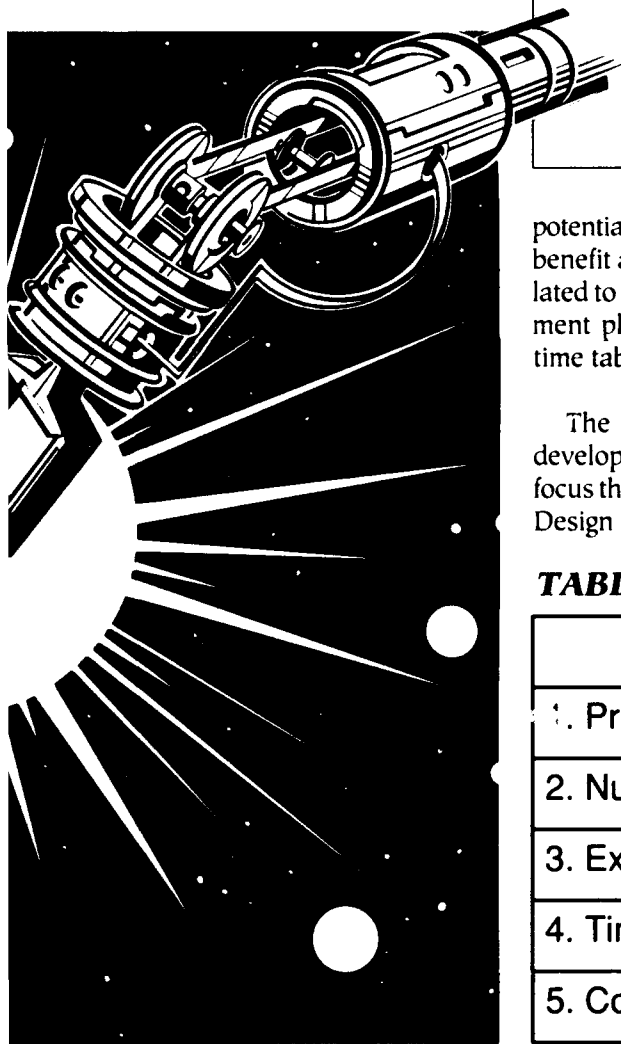
TABLE 1. Project Selection Criteria

| Criterion | Weight (w_i) |
|--|------------------|
| 1. Project Impact on Production Facility | 5 |
| 2. Number of Weapon Systems Affected | 1 |
| 3. Extent of Production Facility Affected | 3 |
| 4. Time Required to Complete Phases II & III | 1 |
| 5. Continuation of Modernization | 2 |

be") facility. Phase I is an essential element of a company's Strategic Modernization Plan (SMP), which includes a projection of future business

majority of the Phase II effort involves accomplishing detailed designs for each selected modernization project, demonstrating proof of concept, perform-

projects to be rated on a common basis. The criteria were applied using a Multiattribute Utility (MAU) analysis approach. Each criterion could be



rated from 0 points (worst case) to 100 points (best case).

When the projects were rank-ordered by total points (which represented the degree of effectiveness), the highly rated, most effective projects were grouped at the top. The least effective ones were at the bottom. Additional criteria were then used to ensure that the selected projects were balanced between degrees of effectiveness, Air Force weapon systems and industry segments. Twelve subcontractor manufacturing modernization programs involving 78 modern-

projects, allowing the determination of which specific aspects and subprocesses were indeed critical in predicting the overall effectiveness of the projects. Eleven key characteristics are listed in Table 2. After analysis, four were found to be critical indicators of success (Table 4). The key characteristics were all important for success. The critical factors were those that clearly differentiated the most successful from the least successful projects.

1. *Maturity of the Weapon System Supported* was a significant factor only to the extent that future production orders could be anticipated.

2. *Stability of the Weapon System Supported* was an important factor. It was especially important for those whose business base was mostly defense related. Size of the Program Supported did not appear significant; nearly every weapon system included in the study was in the \$100s million category.

3. *Business Strategies of the Suppliers* were very important in

with the most commercial work were particularly interested in the applicability of the factory modernization and new process technology to their commercial products. All the companies had a wide production base and were not overly dependent on a single customer. Many had an appreciable business in providing replacement parts and logistic services.

4. *Technical Approach to Top-Down Factory Architecture* in the Phase I study efforts included use of the IDEF₀ methodology in the definition of the baseline "as is" facility and the improved "to be" facility. The most extensive use of IDEF₀ concerned the development of the node tree analysis for production activities. This tree was commonly used as a framework to analyze cost and schedule drivers. A typical issue encountered was the lack of sophistication and wide diversity of a company's cost-capturing systems which often led to the use of "engineering estimates" for the assignment of costs to the various nodes. Many companies were surprised by some of the results. In several cases, this precipitated a much more extensive modernization effort than was initially planned.

Other system engineering approaches were employed as well. These included: simulations, facility layouts, value analysis techniques, flow charts, activity-based costing, and product cost decompositions. Most of the time, these were accomplished by specialized tools used by consultants.

In nearly every case, an attempt was made at the beginning of Phase II to show traceability to the Phase I analysis. However, the degree of traceability ranged from excellent to marginal. Surprisingly, most companies, prior to IMIP, had done minimal modernization planning other than replacing items of equipment, such as machine tools. The Phase I study was essentially their first significant opportunity to consider the future of their facility, and many thought that

TABLE 2. IMIP Key Characteristics

| | |
|-----|---|
| 1. | Maturity of the Weapon Systems Supported |
| 2. | Stability and Size of the Weapon Systems Supported |
| 3. | Business Strategy (Company) |
| 4. | Technical Approaches to Top-Down Facility Architecture |
| 5. | Requests for Proposals/Phase I Contract and SOW Adequacy |
| 6. | Use of Supporting Consultants |
| 7. | Adequacy of Phase II |
| 8. | Adequacy of Phase III |
| 9. | Time to Accomplish IMIP Phases I and II |
| 10. | Contractor Executive Management Commitment to Factory Modernization |
| 11. | Business Characteristics (Demographics) of the Subcontractor |

ization projects were selected for detailed evaluation.

Key Characteristics

Key aspects and subprocesses of the selected projects served as a framework for organizing and evaluating the collection of data. Key features were compared and contrasted for the most-effective and the least-effective

understanding the business climate in which the companies operated. Every supplier had mostly firm-fixed-price contracts. Some had multiyear contracts, but did not view them as important as becoming a "preferred" supplier to a prime. Most of the companies had more than 10 percent of their business in the nongovernment, commercial sector, and were forecasting this percentage to increase. Those

this effort alone was extremely worthwhile. Many companies continued to maintain and update the Strategic Modernization Plan after the IMIP Phase I was completed.

5. The effectiveness criteria for *Phase I RFP, Contract, and SOW Adequacy* were based on evaluating how well the contracting process communicated the government's requirements. In all cases, these aspects were found to be satisfactory.

6. Nearly every project made *Use of Supporting Consultants* during Phase I. Typical Phase I activities performed by consultants included development of the top-down facility analysis, team training, evaluation of baselines, analysis of candidate projects, and technical approaches for candidate projects. The type of consultants used ranged from assistance provided by prime contractors, to using a corporate team of experts (for those companies that were part of larger corporate family), to specialized consultants selected through a competitive process. Issues raised about using consultants were: (a) Some were not expert in specific details of the company's business, so a period of time was required to "bring them up to speed"; (b) a tendency to force-fit all problems into their methodologies and tool set even when they were not appropriate; and (c) too extensive an involvement in evaluating and selecting improvement projects during Phase I was not effective because the knowledge tended to leave with the consultant—correspondingly, the company's personnel did not have a high degree of ownership of the results.

Phase I programs that did not use consultants were generally less effective than those that did. The amount of effort contributed by consultants varied significantly. The most important factor was the way in which consultants were used. When they supported an in-house team and had a clearly defined and understood role, they contributed effectively. When

consultants were hired to "fill a square" and were not incorporated as part of the company's team, their contribution was markedly reduced.

7. *Adequacy of Phase II* was based on an examination of the quality of the effort. As would be expected, the adequacy of Phase II paralleled the effectiveness ratings. The more effective efforts accomplished Phase II projects within the resources that were planned; although, in many cases Phase II projects took somewhat longer to complete than originally planned. In almost every instance, time and effort to develop new software was greatly underestimated. This was true for software to control machine tools and for software that supported facility information systems.

In general, the smaller companies tended to overestimate the number of simultaneous projects that could be managed effectively using their normal management style. Those projects that completed Phase II development on the same hardware/software systems that would be used in production were more effective, even though this approach required early capital investments.

Projects terminated during Phase II fell into three categories: (A) technically successful but not economically viable; (B) technical failures; and (C) technically successful/usable, but superseded by a better, less-expensive alternative.

For Case A, lack of a reasonable payoff did not necessarily mean there was insufficient payoff for the Department of Defense as a whole, but for the sponsoring Air Force program office. The lack of a commonly accepted standard approach to define savings of "intangible" benefits, like improved quality, reduced indirect costs, shorter lead times, and so on, led to unfavorable return-on-investment (ROI) using a discounted cash flow model even though overall benefits were thought to be significant.

There were examples in which the Phase II analysis determined there was insufficient payoff to proceed.

TABLE 3. Average Age of Equipments for IMIP Companies

| Company | Average Age of Product Equipments [before -> after] |
|------------|---|
| Company 1 | Not meaningful. |
| Company 2 | 24 yrs -> 3 yrs |
| Company 3 | 20 yrs -> 6 yrs |
| Company 4 | Large range, old and new. |
| Company 5 | Wide range. |
| Company 6 | 15 yrs -> 8-10 yrs. |
| Company 7 | 3 yrs (new facility). |
| Company 8 | Not meaningful. Upgraded old. |
| Company 9 | Not available |
| Company 10 | 14 yrs -> 8 yrs. |
| Company 11 | unknown -> 8 yrs |
| Company 12 | Not available |

These instances were due to poor benefits definition, inadequate definition during Phase I, or a reduction in Department of Defense orders.

For Case B, the less effective programs tended to "throw in the towel" when unexpected and complex technical challenges arose, while the more effective programs solved the challenges. This is indicative of the ground-

TABLE 4. Critical Characteristic Analysis

| Effectiveness Rating | Business Strategy Percent Commercial Business | Technical Approaches to Phase I | Phase I Adequacy Main Thrust of Modernization Effort | Contractor Management Commitment Multidiscipline, Full-Time Team |
|----------------------|---|---------------------------------------|--|---|
| A | 50 to 70 | Facility-wide | Factory-wide modernization | Yes |
| A | 50 to 70 | Facility-wide | Enterprise Information System | Yes |
| A | 10 to 25 | Facility-wide | Quality and Mfg Info System | Yes |
| A | 10 or less | Facility-wide | Specific process Improvements | Yes, except Engineering |
| B | 10 to 25 | Facility-wide | Manufacturing Information Systems, Quality, Auto Assembly | Yes |
| B | 10 to 25 | Product Specific | Mfg Info Systems and Indirect, Process Upgrades | Yes, except Manufacturing |
| B | 10 to 25 | Facility-wide | Quality | Yes |
| C | 50 to 70 | Product Specific | Specific Process Improvements, not Automation | Yes (PM only Full Time) |
| C | greater than 70 | Equipment Specific | Specific Automation Projects | At start, not continued |
| C | greater than 70 | Operation Specific | Manufacturing Information Systems | At start, not continued |
| C | 10 or less | Operation Specific | Quality and Auto Assembly | At start, not continued |
| D | 10 or less | Operation Specific | Specific Automation | At start, not continued |

work laid during Phase I. Less-effective programs tended to orient the Phase I analysis toward a high degree of automation, and took an overly simplistic view of what it really takes to automate a process. The more effective programs used a more pragmatic approach for automation. Government policies that emphasize capital investment tend to "push" subcontractors toward a higher degree of automation than may be worthwhile.

Case C was usually caused by incomplete technical analyses performed during Phase I. For example, while a project was taking one approach to greatly improve the repair procedure for an expensive metalworking die, another group in the same company was changing the die design to eliminate the cause of the failure mode. This was symptomatic of many projects that fell in this category; namely, there was insufficient consideration in Phase I to address the root cause of the

identified problems and to trade-off solutions.

8. *Adequacy of Phase III* was assessed by evaluating how closely implementations followed the plan and produced expected benefits. In many cases, the Phase III implementations took longer than estimated. This was usually due to the pressures of instant business. It was almost always more important for the company to satisfy current orders than to stop the production process or to take key personnel away from a key job to implement improvements.

9. *Time to Accomplish Phases* revealed that most programs experienced large time gaps between phases. Equally distressing was the stretch-out of Phase II programs due to inadequate funding rates. The length of Phases I and II, and the gap between the two phases, for the 12 IMIPs ranged from a few months to more than 24

months. Sequential funding of multiple Phase II projects at a subcontractor's facility led in some instances to the modernization program stretching over many years. Gaps between phases had an especially severe impact at small companies which found it difficult to return to a high level of enthusiasm and commitment after a protracted wait.

10. Assessing the *Contractor Executive Management Commitment* was measured by the formation of a multidisciplinary team. The most effective programs had a full-time team led by a senior manager who reported directly to the president or chief executive officer. Team members were from different organizations within the company and represented a spectrum of appropriate disciplines.

Another measure of the management's commitment was the amount of capital investment made

by the company. In this respect, comparing the average age of equipments before IMIP with that after modernization implementation can be enlightening. Table 3 summarizes the data collected during this study. Companies are listed in order of effectiveness.

Dramatic reductions in average age of equipment are associated with the modernization projects that were rated high on effectiveness. Not all capital investments included within this study were directly due to the modernization projects, but many investment decisions were significantly influenced by the facility analysis.

In some cases the average age of equipment was not meaningful due to several factors. Phase I analysis, in some instances, pointed out the high cost of maintaining multiple plant sites; these were consolidated and older equipment surplused. Thus, the average age decreased but not due to capital expenditures.

The age of an item of equipment, such as a machine tool, is determined by its acquisition date. Thus, the age of record is not changed even when the equipment is upgraded.

Several factories had a wide range of equipment types such as vacuum furnaces and microelectronic assembly stations. The companies had modernized the equipment that was crucial to the manufacture of its high technology products; only moderate upgrades were made to the remaining equipment. This led to a bimodal distribution of equipment age.

11. *Business Characteristics of the Subcontractor/Vendor* is the demographic information about the companies. Sizes of companies varied by an order of magnitude from a low of \$45 million to a high of \$490 million annual sales. The industry sectors represented by the 12 companies included electrical and electronic equipment, metalworking, casting, metal products and aerospace manufacturing.

The most important factor in the success of a program is a consistent and continuing top-level company management commitment.

Critical Characteristics

Of the key characteristics examined, four were found to be *critical*. In this context, "critical" is defined as a characteristic that could predict and influence effectiveness of the modernization effort to a significant degree. A synopsis of the critical characteristic data is presented in Table 4. Effectiveness ratings are given in letter grades ranging from A (very effective) to D (not very effective). This was done to group those programs with similar numerical scores. The four critical characteristics follow.

—*Business Strategies (Company)* as indicated by percent commercial business. (Other metrics of this characteristic generally followed the same trend.) Typically, a company needed commercial business to realize an attractive return-on-investment for capital expenses. Government incentives alone were perceived to be unreliable or inadequate. The two least-effective programs had less than 5 percent commercial business and thus did not have sufficient commercial business to generate the additional required payoff.

—*Technical Approaches* to top-down facility analysis, as indicated by the technical approach to modernization. The first was the *traditional* approach of quickly zeroing in on obvious, specific improvements relating to particular

operations, pieces of equipment, or subassemblies/products. This approach resulted in projects that ranged in effectiveness from above average to much below average. The other approach started with a *comprehensive facility-wide* analysis to establish a framework for judging the relative payoff of improvements from a "systems perspective" of the entire enterprise. The facility-wide analysis approach produced the most-effective efforts. For example, a company with a highly-effective program had a prior modernization strategy that planned to replace old equipment. However, as a result of the Phase I analysis, a decision was made to proceed with the complete revamping of the facility, which included substantial rearrangement of the process flow. Nearly the same equipment was acquired, but it was employed more effectively than originally envisioned.

—*Phase I Adequacy*, as indicated by the focus of the analysis; i.e., enterprise-wide, quality, MIS, process specific improvements, etc. (Adequacy of the Phase I plan is related to the technical approaches critical characteristic.) Those which only emphasized specific process improvements were not necessarily failures but were not as effective, in general, as those which made system-wide improvements in conjunction with process improvements. Using the systems approach, important considerations for "above-the-shop floor" processes (e.g., application of group technology, material handling improvements, and maintenance of the software for automated operations) were adequately addressed. In essence, this critical characteristic is the assessment of how well the technical approach was performed.

Contractor Executive Management Commitment to facility modernization, as indicated by the *formation of a full-time, multidisciplinary team*. The commitment of company senior management is imperative. Programs with strong management backing were able

to overcome obstacles including technical surprises, gaps in government funding, and incentives which were less than anticipated.

Lessons Learned

Future manufacturing modernization programs can incorporate the best practices of past programs to reduce their risk and cost. This section summarizes what worked particularly well and what should be avoided.

Lesson Learned 1: Successful Manufacturing Modernization Programs Require High-Level Corporate Commitment

—The most important factor in the success of a program is a consistent and continuing top-level company management commitment. All sources of information confirmed the necessity for senior management to place necessary emphasis on the program, and to ensure availability of resources to plan, analyze, develop and execute the program. As problems arose during the conduct of the program, those with the corporate management commitment managed to overcome them to a far greater extent than those that did not have such a commitment.

—Smaller, privately-held companies gained top-management support for development of a strategic modernization plan (SMP) in a relatively straightforward manner. Publicly-held companies, particularly members of a corporate family, required additional time to obtain upper-level company management commitment.

—The first concrete sign of a serious commitment was the formation of a multidisciplinary team at the start of Phase 0. An important outcome of this activity was development of specific goals regarding modernization opportunities and expected benefits to company and customer. In general, potential contract funding support, and incentives for future savings were significant enticements for

management, even when capital spending approval had not previously been forthcoming.

The most effective programs resulted in wide-ranging changes within the company. However, corporate culture takes time to change and requires involvement by all levels of the work force. The commitment by senior management helps promote this flow-down throughout the company. Benefits of involving the entire company included continued improvement through education and training, development of a wealth of improvement ideas, a wide base of support for implementation and pride of ownership in results.

—Some less-successful modernization projects had top-level commitment, but had an appreciable amount of the Phase I effort accomplished by consultants who were not under the direction and support of an in-house team. Unfortunately, little internal interest at the "working level" was developed to pursue projects once the consultants left. Technical and/or implementation issues tended to become obstacles and often were used as reasons to cancel projects. On the other hand, companies which established a strong in-house team supplemented by outside expertise were generally successful.

—Usually the subcontractor's first in-depth exposure to factory modernization was through a prime contractor. The subcontractor's executives typically viewed modernization first as an opportunity to enhance their business relationship with the prime, possibly by becoming a preferred supplier. The opportunity to improve their competitive advantage by modernizing facility operations was a strong additional incentive.

Lesson Learned 2: Strategic Modernization Plan Critically Important

—Highly effective modernization was accomplished by companies that

developed long-range strategic plans for improvements addressing concrete needs. Plans affected direct and indirect facility areas in an integrated fashion so that individual areas were not suboptimized. A sound technical approach using a variety of engineering analysis tools guided by a system engineering approach was necessary to *attack root causes of problems*, not symptoms. For example, a subcontractor with an effective modernization program had a key supplier with a persistent quality problem. Rather than automating receiving inspection, the subcontractor worked with the supplier's manufacturing process to attack the root cause of the quality deficiency.

—Less-effective programs rapidly converged on improvement projects which had benefits that could be quantified for a particular production process, but which were not the most important from a strategic viewpoint. For example, one program focused on automating manual operations that accounted for only a small percentage of manufacturing costs and quality problems, while costly operations were ignored because they could not be easily automated.

—Prioritization of efforts ensures that the most important projects are accomplished first. This is facilitated by a comprehensive facility-wide analysis as part of a company's strategic plan. The most effective strategic plans ensured that the modernization projects fit together realistically. For example, a successful company that had a strategic goal of reducing work-in-process (WIP), established a comprehensive plan for just-in-time (JIT) inventory control that included rearrangement of machine tools, upgrades to equipment, improvements to the factory information systems, training for shop-floor personnel and production planners, and incentive arrangements with their suppliers.

—Improper use of discounted cash flow models during Phase I led to

erroneous management decisions regarding the true viability of improvement projects. Typical sources of error included cost data of dubious accuracy, overly optimistic business projections, and underestimation of implementation costs. Inevitably, these mistakes became obvious during Phase II after significant time and effort had been expended on ill-fated projects with little, if any, chance of implementation.

—Over-reliance on discounted cash-flow models to prioritize projects created the following types of issues: (a) overemphasis on the automation of manual operations due to an apparently easy determination of savings from reduced direct labor and reasonably good estimate of capital investment needs; and (b) neglect of the major savings potential in the indirect areas that often required little capital investment.

—Effective modernization programs maintained an implementation focus. Projects were pursued that could be readily implemented. If some difficulty was anticipated, a mitigation strategy was developed. *Resistance to change* was an implementation barrier highlighted by nearly every company. Successful companies established a continuing dialogue with the work force explaining the changes and the expected results. They followed through with necessary training.

—Effective modernization programs became part of the company's overall strategy to become a "world class manufacturer." One company established a special position of "world class manufacturing manager." This infused the entire company with a dedication toward manufacturing excellence that transcended the original goals of the modernization effort.

Lesson Learned 3: Significant Obstacles to Implementation Exist

—Time delays for approvals and contractual actions were responsible

The lessons learned identified during this study provide a clear blueprint for improving future facility modernization projects.

for loss of effectiveness in the IMIP programs at the subcontractor level. Unilateral revision of the rewards, for whatever reason, was a negative factor in the implementation decision.

—In instances where modernization projects impacted product lines across multiple weapon system programs, implementation planning did not always allocate the time or effort for changes in engineering technical data to be made since the scope was too narrowly viewed as being manufacturing-based. Sizeable productivity payoffs were missed particularly when significant changes to the quality verification methods were accomplished by a facility modernization project, but could not be implemented on the production floor because not all affected primes had an opportunity to evaluate adequacy of the change for their components.

Summary

The health of the defense industrial base is a vital ingredient in our country's total military capability. This industrial base includes many subcontractors and suppliers who provide critical support to prime weapon system contractors and follow-on support to logistic centers. Their overall health encompasses financial and up-to-date manufacturing capability. It is very important that the time and resources devoted to modernizing their facilities are spent effectively.

The lessons learned identified during this study provide a clear blueprint for improving future facility modernization projects. They include: required corporate commitment by senior management; a company's strategic modernization plan to be critically important as the framework for improvements; and (3) mitigating obstacles to implementing projects through early involvement of key people.

Endnotes

1. *Study of the Industrial Modernization Incentives Program*, Final Report, Volume I, SDRL Sequence 002, Contract F33615-89-C-5708, Task Order 24, Submitted by Technology Assessment and Transfer, Inc., Annapolis, Md., in support of WL/MTX, Wright-Patterson AFB, Ohio, April 19, 1991.
2. Robert E. Schafrik, "Encouraging Competition at the Subcontractor Level," *Program Manager*, Sept.-Oct. 1984, pp. 28-31.
3. Discussions with Mr. Jim Webber, HQ AFSC IMIP program manager.
4. Discussion with Mr. William Harris, IMIP Technical Manager, Air Force Manufacturing Technology Directorate, WL/MTX, Wright-Patterson AFB, Ohio.
5. Softech, Inc., *Integrated Computer Aided Manufacturing (ICAM) Function Modeling Manual (IDEF₂)*, Report Number UM110231100, prepared for Air Force Manufacturing Technology Division, Air Force Materials Laboratory, Wright-Patterson AFB, Ohio, Contract F33615-78-C-5158, June 1981.
6. David Marca and Clement McGowan, *SADT-Structured Analysis and Design Techniques*, McGraw-Hill, 1987.

THE ARMY REUSE CENTER— COST AVOIDANCE THROUGH SOFTWARE REUSE

Software Development Support for Program Managers

Marrea Riggs
Director, Army Reuse Center

Program managers are under constant pressure to provide their users with top-notch products on time and within budget. The Army Reuse Center (ARC) staff understands those requirements. The ARC was established to support the development and fielding of reliable, high-quality systems, reduce the time and resources required to develop those systems, and reduce software life-cycle costs. These objectives can be achieved through the effective application of software reuse.

The ARC is in the business of working with program managers to incorporate software reuse into each phase of the development life cycle. The ARC provides, through its effective, automated library system, an array of high-quality, reusable software components and a range of reuse engineering support. Its mission is to make planned, systematic reuse an integral part of the Army's day-to-day development efforts. The ARC's approach addresses three key areas:

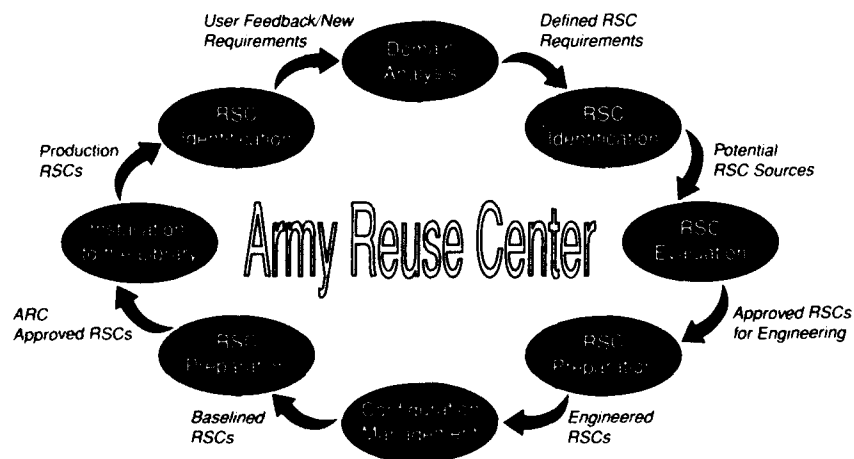
- Sound reuse policies and procedures
- Customer support and training
- Automated library system.

Sound software engineering practices are the hallmark of the ARC's Software Reuse Program. Each key procedure is fully documented to ensure high-quality reusable software components (RSCs) and repeatable processes. The procedures are domain analysis; reusable software component identification, evaluation and preparation; configuration management; configuration manage-

ment life cycle, thereby institutionalizing software reuse as part of the organization's development process. This integration marks the beginning of a change in the way software is developed—from "line-by-line" to "component-by-component."

The ARC provides direct support and hands-on training in domain analy-

FIGURE 1. RSC Certification Process



ment; quality assurance; and library population and user extraction.

These reuse procedures can be fully integrated into the software develop-

ment life cycle, thereby institutionalizing software reuse as part of the organization's development process. This integration marks the beginning of a change in the way software is developed—from "line-by-line" to "component-by-component."

and differences among related systems and provide the basis for the sharing of reusable software components (Figure 1). The ARC's domain analysis process will evolve as it is repeatedly applied within Army and DOD reuse efforts, and in response to technology advances. This process is being used and validated as part of ARC's FY92 reuse initiative within the Program Executive Office, Standard Army Management Information Systems (PEO-STAMIS).

On-Site Support and Training

The ARC staff can provide on-site support and training in selection, comparison and download of a range of reusable software components. These components will have been classified, certified and evaluated in terms of reusability, complexity and other qualitative measures.

Other support includes reuse planning, reuse engineering support and other program/project assistance. The ARC staff can work within specific domains, or "families" of programs, to develop generic architectures and domain models; to identify specific reuse opportunities and high-demand components; identify and recommend potential donor-client matchups; and develop reuse implementation schedules. The staff can work with individual programs to assist in mapping system/generic architectures; incorporating software reuse into the design; locating and extracting reuse components; integrating reusable components; and reengineering existing systems.

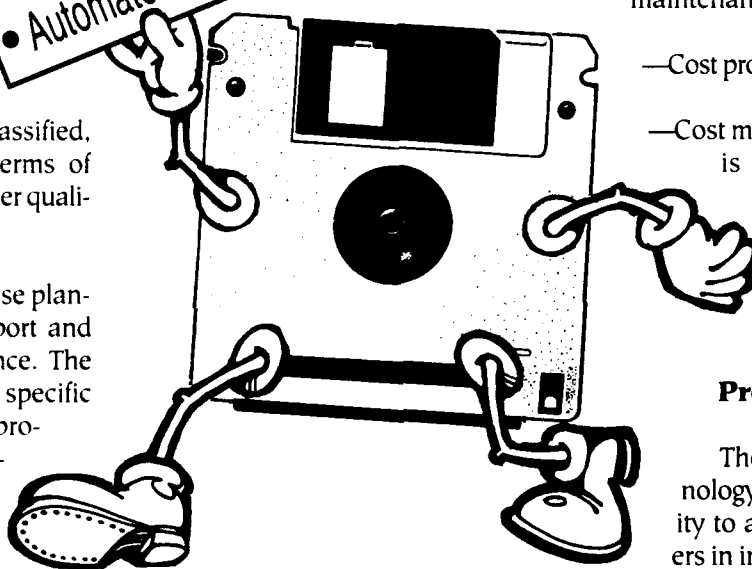
The ARC library system is the cornerstone of the ARC operation. Much more than a repository, it is an interactive library system that will support users in analysis, identification and retrieval of reusable software compo-

nents. The library system guides ARC engineers in application of rigorous software engineering procedures, strict configuration management, and high-quality assurance during component certification and maintenance. A total of 1,401 reusable software components, representing more than 1 million lines of code, have been certified and are available within the library.

Reusable Ada Components

In addition, ARC personnel are certifying and installing a large number of reusable Ada com-

- Sound reuse
- Customer support
- Automated library



ponents from the Army Worldwide Military Command and Control System (WWMCCS) Information System (AWIS) Program. The AWIS has increased the portability of its application software through utilization of a layered architectural approach. The AWIS is providing the ARC with its applications support layer, which contains common service interfaces for data base, graphic user interface, external interface manager and various utilities. Incorporation of this common service layer (or some major portion of it) within other systems having similar requirements and standards would provide significant benefits in terms of cost avoidance or cost sav-

ings.

As software development costs continue to skyrocket, it will become increasingly imperative that program managers evaluate, and pursue, the opportunities for software reuse within their development efforts. Reuse benefits can include:

- More reliable and higher-quality systems
- Software already tested and certified by the ARC
- Faster delivery of system software
- Lower software development and maintenance costs
- Cost projections more accurate
- Cost models show reuse code is much cheaper than developed code
- System easier to upgrade if reuse planned in.

Proven Capability

The ARC has the technology and proven capability to assist program managers in integrating reuse within their software development process. Supporting the Department of the Army and DISA/CIM, the ARC will continue to provide an expanding range of reuse services that include domain analysis, reuse engineering support, reuse familiarization and training, and component certification and reengineering.

For more information or to obtain a copy of the ARC General Information or New User Packet write or call:

Army Reuse Center, USAISSDCW,
Attn: ASQB-IWS-R, Stop H-4, Ft.
Belvoir, VA 22060-5426, (703)
285-6272, DSN 356-6272, Fax (703)
285-6377.

EVOLUTIONARY ACQUISITION AND DOD'S REVISED ACQUISITION STRATEGY:

The Defense Systems Management College— DOD'S Agent For Change

Edward Hirsch
Brigadier General, U.S. Army (ret.)

When Mr. Donald J. Yockey, the Under Secretary of Defense for Acquisition, announced the Department's new acquisition strategy on May 20, 1992, the event marked the end of intensive developmental planning efforts. It also was the beginning of equally intensive efforts to ensure the strategy would be implemented effectively and efficiently.

The new strategy recognizes the significant recent changes in the national security environment. It emphasizes research, development, Advanced Technology Demonstrations, technology insertion, continuous improvement of existing systems and components—and the fact that there will be fewer new major systems produced in fewer quantities than in the past. The cooperation of the three major players in this new and flexible acquisition process is essential to its

Mr. Donald J. Yockey at right.

General Hirsch is a member of the Executive Institute and Corporate Board of the Defense Systems Management College.

The new strategy...emphasizes research, development, Advanced Technology Demonstrations, technology insertion, continuous improvement of existing systems and components—and the fact that there will be fewer new major systems produced in fewer quantities than in the past.



success. The executive and legislative branches of government must work together and in harmony with that most essential contributor to the acquisition process, industry.

Team Building to Make It Work

The required level of cooperation among these three stakeholders may not yet have been achieved; however, real progress has been, and is being made toward that end. Two important initiatives, reflective of this spirit of cooperation, appeared in the FY 1991 Authorization Act (PL 101-510).

The first, Section 800, directed DOD establish an Advisory Panel on Streamlining and Codifying Acquisition Laws under the sponsorship of the Defense Systems Management College. The Panel has as its purpose to:

(1) Review acquisition laws with a view toward streamlining the acquisition process; (2) Recommend the repeal or amendment of laws the Panel feels are unnecessary for the establishment and administration of buyer and seller relationships in procurement; and (3) Prepare a proposed code of relevant acquisition laws. This effort is characterized by the active cooperation and personal participation of members of Congress, key staff personal, DOD, senior representatives from industry, the legal profession and academia. It has the potential to eliminate or reduce what appears to many as the principal legislative roadblocks to true cooperation among the three participants in the acquisition process.

The second legislative initiative is Section 809 of the same public law. This section authorizes DOD to identify up to six major programs as Pilot Programs to conduct business in accordance with standard commercial, industrial practices. The DOD may request a waiver or limitation of any provision of law it considers appropriate. This on-going action also enjoys the complete cooperation of the three stakeholders in the process. It has the potential to not only increase the effectiveness and efficiency of the acquisition process, but to enhance it by exploiting commercial practices to a greater degree than is now the norm.

However, successful implementation of the new strategy demands increased cooperation among the participants.

No Need for a Paradigm Shift

The principal imperatives behind the previous acquisition process were



Command and Control

An evolutionary acquisition strategy is well-suited to high technology and software intensive programs where requirements beyond a core capability can generally, but not specifically, be defined.

the threats perceived to be generated by the existence of the Soviet Union and the need to cope with them—quickly. That process, developed over time, proved to be adequate to the task. Essentially, it assumed that if a major defense weapons system was approved for Concept Demonstration, it would ultimately be approved for production. The new flexible acquisition strategy continues to support that view as well as the requirement for full funding of the program. It also acknowledges the possibility of program termination prior to production and subsequent diversion of program funds to other activities. The emphasis shifts from production to meet an all-important, time critical

initial operational capability, to reducing risks by increased use of advanced technology demonstrations prior to the Demonstration/Validation Phase.

However, increased emphasis on risk reduction activities is demanded throughout the acquisition process as is the early and continuous involvement of the user with the acquiring community. This emphasis constitutes a subtle but significant change which impacts the stakeholders in the process. Fortunately, no new legislation is required to accommodate this new strategy. If the legislative actions required and authorized by PL 101-510 come to fruition, the legal basis for the acquisition process will be simplified and facilitated. Congress needs only to continue to be receptive, understanding and responsive to the changing needs of DOD and industry as they strive to operate within the changing environment. The Department of Defense and industry on the other hand, operating within the policy guidelines established by the Department's 5000 series of directives and instructions, must exploit the flexibility inherent in those guidelines and modify their approach to acquiring many types of weapons systems. The template for such modification exists and is currently identified in the 5000 series as Evolutionary Acquisition (EA).

Evolutionary Acquisition—Permitted, Not Preferred

The 5000 series of DOD directives and instructions wisely devotes substantial space to define and then require the application of the concept of Evolutionary Requirements Definition. The series recognizes that operational performance requirements for a new acquisition start must evolve from broad operational capability needs to detailed system-specific performance requirements. However, DODD 5000.1 does not identify or

define EA. It is only in DODI 5000.2 that EA is addressed as an alternative acquisition strategy that should be considered for systems where requirements refinements are anticipated or where a technology risk or opportunity discourages immediate implementation of a required capability. It states that:

Evolutionary acquisition is an approach in which a core capability is fielded, and the system design has a modular structure and provisions for future upgrades and changes as requirements are refined. An EA strategy is well suited to high technology and software intensive programs where requirements beyond a core capability can generally, but not specifically, be defined. This approach is described in Joint Logistics Commanders Guidance, *Evolutionary Acquisition, An Alternative Strategy for Acquiring Command and Control (C2) Systems*.

In essence, DOD requires an evolutionary approach to requirements generation, and permits an evolutionary approach to the development and acquisition processes. The new acquisition strategy creates an environment in which EA can not only make a significant contribution to enhancing effectiveness and efficiency of the process, but can become the preferred approach to acquiring most weapons systems.

Evolutionary Acquisition— The Joint Logistics Commanders View

In March 1987, the JLCs in cooperation with the Defense Systems Management College, published their guidance recommending the use of an EA strategy in acquiring C² systems. They further stated that while their guidance was aimed specifically at the use of an EA strategy in acquiring C2 systems, "...the principles discussed may also be applicable to the acquisition of other kinds of systems...." In May 1992, the JLCs reaffirmed that

view. In a letter to the Commandant, they requested that DSMC work with them to update the 1987 document and expand the guidance on EA to include as many types of defense systems as possible. Clearly, during the five years that elapsed since they originally endorsed EA as a viable and recommended strategy, the JLC's were convinced that it was equally viable for systems other than command and control.

Evolutionary Acquisition— Now More Than Ever

The Department's revised acquisition strategy clearly states the requirement for a more robust science and technology program which can produce mature technology and innovative ideas related to manufacturing processes. It is equally clear in its demand that acquisition-system activities be undertaken only when: "(1) The technologies have been demonstrated, thoroughly tested, and shown to be producible; (2) There is a clear and verified military need for the new system or system upgrade; (3) The new system or system upgrade is cost-effective." These requirements dictate a new, close and continuing relationship among the science and technology community, the developer, the tester, the supporter and the user.

The EA approach was conceived to develop and nurture just such a relationship. It is comprised of the following key elements:

- A concise statement of operational concepts and requirements for the full system.
- A general description of the functional capability desired for the full system.
- A flexible, well-planned overall architecture, to include process for change, which will allow the system to be designed and implemented in an incremental way.

- A plan for incremental achievement of the desired total capability.
- Early definition, funding, development, testing, fielding, supporting and operational evaluation of an initial increment of operational capability.
- Sequential definition, funding, development, testing, fielding, supporting and operational evaluation of additional increments of operational capability.
- Continual dialogue and feedback among users, developers, supporters and testers.
- Use of the "Build a little, Test a little, Field a little" development philosophy, which should result in timely fielding and support of each increment of operational capability.
- Use of flexible system architecture, which should facilitate evolutionary enhancement and expansion of the system while the mission continues to be supported.

EA Meets All the Demands Of the Revised Acquisition Strategy

- EA operates within the existing acquisition process—it conforms to the 5000 series of directives and instructions.
- EA is an event-driven, not a calendar-driven approach.
- EA is characterized by a "Show-Me" attitude that results in maturity early-on in the acquisition cycle.
- EA requires full funding for each increment of operational capability prior to approval for development.

- EA supports the view that older, capable platforms should be upgraded with advanced components to enhance mission accomplishment.
- EA, properly applied, delays system obsolescence.
- EA is designed to answer the three questions critical to all acquisition programs—Do we need it? Will it Work? Is it cost effective?
- EA can provide industry a pay as you go capability in design and development without dependence upon getting well in production.

EA Places Unique Demands Upon the System

The key ingredient of an EA development and acquisition is the continuous, combined user-developer-tester-supporter effort that must exist throughout the system life cycle. It provides the feedback loop essential to success. Any reluctance or inability to communicate and cooperate on the part of even one contributor to the effort can generate almost insurmountable problems. The challenge to the testing community is particularly daunting. The testers must become involved early in the process, work closely with the users and developers and still maintain their objectivity and the integrity of their own process.

Acquisition leaders at the senior decision-making levels must recognize that programmatic decisions made early in the program life cycle are generally the most critical in their impact upon cost, schedule and performance. It is essential that high quality, experienced personnel are assigned to whatever entity is established to plan and execute the as yet nonexistent program in its earliest stages. The normal approach of assigning junior personnel who can be made readily available on a temporary basis is unacceptable. The best must be chosen early and assigned for the long term



Evolutionary acquisition is an approach in which a core capability is fielded, and the system design has a modular structure and provisions for future upgrades and changes as requirements are refined.

The contracting community must exercise the flexibility inherent in our system and recognize that unless something is expressly prohibited, it is permitted. Competition is an important element of the process; however, when a loyal and quality supplier is available, his services must not be supplanted by a low bidder whose past performance and quality is questionable or unknown.

All of these demands are attainable. Evolutionary acquisition can facilitate the accomplishment of the mission for which the revised acquisition strategy was designed. Evolutionary requirements generation, combined with evolutionary acquisition can provide the preferred approach

to achieve the new strategy's objectives.

EDITOR'S NOTE:

Evolutionary Acquisition, An Alternative Strategy for Acquiring Command and Control (C²) Systems is being revised.

FOREIGN LANGUAGE

Instruction Contract

The CACI Language Center has been awarded a contract from the Defense Language Institute (DLI) to provide foreign language instruction services to Department of Defense personnel in 35 languages, such as: Albanian, Arabic, Bulgarian, Chinese, French, German, Japanese, Lingala, Hebrew, Portuguese, Spanish and Turkish.

The CACI language training programs enable American diplomatic and military personnel to perform their duties effectively overseas by imparting foreign language skills and essential cultural information. According to Paula Satisfsky, Director of CACI's Language School, "We not only teach the language, we teach our students to think like native-speakers. By training DOD personnel to function optimally in foreign cultures, we promote clear communication between the United States and other countries on matters affecting international security and world peace."